



HAL
open science

Modeling the internet of things in configurable process models

Kunal Suri

► **To cite this version:**

Kunal Suri. Modeling the internet of things in configurable process models. Modeling and Simulation. Université Paris Saclay (COMUE), 2019. English. NNT : 2019SACLL005 . tel-02059346

HAL Id: tel-02059346

<https://theses.hal.science/tel-02059346>

Submitted on 6 Mar 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Modeling the Internet of Things in Configurable Process Models

Thèse de doctorat de l'Université Paris-Saclay
préparée à Télécom SudParis et au
Commissariat à l'Énergie Atomique et aux Énergies Alternatives (CEA)

Ecole doctorale n°580 Sciences et Technologies de l'Information et de la
Communication (STIC)
Spécialité de doctorat : Informatique

Thèse présentée et soutenue à Télécom SudParis, Évry, le 11/02/2019, par

M. KUNAL SURI

Composition du Jury :

M. Nazim Agoulmine Professeur, Université d'Évry Val d'Essonne	Président
Mme. Chirine Ghedira Guegan Professeur, Université Jean Moulin Lyon 3	Rapportrice
M. Michael Mrissa Professeur, Université de Pau et des Pays de l'Adour	Rapporteur
Mme. Maude Manouvrier Maître de Conférences, Université Paris-Dauphine	Examinatrice
Mme. Wided Guédria Chercheur, Luxembourg Institute of Science and Technology	Examinatrice
M. Walid Gaaloul Professeur, Télécom SudParis	Directeur de thèse
M. Samir Tata Professeur, Télécom SudParis	Co-encadrant
M. Arnaud Cuccuru Ingénieur de recherche, Commissariat à l'Énergie Atomique et aux Énergies Alternatives	Co-encadrant

Titre : Modélisation de l'Internet des objets dans des modèles de processus configurables

Mots clés : Internet des objets ; Processus Configurables ; Modélisation ; Gestion des Processus

Résumé : Un nombre croissant d'entreprises internationales ont adopté les systèmes d'information centrés-processus pour profiter des avantages de l'utilisation de processus rationalisés basés sur des modèles prédéfinis, également appelés modèles de processus métier. Cependant, l'environnement commercial dynamique actuel exige de la flexibilité et la réutilisation systématique des processus métier, qui se manifeste par l'utilisation de modèles de processus configurables (CPM). Ceci évite le développement de processus à partir de zéro, qui est à la fois une démarche fastidieuse et sujette à de nombreuses erreurs, et facilite le partage d'une famille de variantes de processus métier pouvant être personnalisées en fonction d'exigences métier concrètes. Par ailleurs, l'adoption des ressources de l'Internet des objets (IoT) dans les processus d'entreprise inter-organisationnels est également en croissance constante. Cependant, ces ressources IoT doivent être utilisées efficacement. Ces dispositifs IoT sont hétérogènes en raison de leurs propriétés et de leurs fabricants (normes propriétaires), ce qui pose des problèmes d'interopérabilité. De plus, étant limitées, elles doivent être allouées (et consommées) en gardant à l'esprit des contraintes, tels que le coût énergétique, le coût de calcul, etc. pour éviter les pannes pendant leurs consommations par les processus. Il est donc essentiel de modéliser explicitement la perspective des ressources IoT dans les modèles de processus métiers lors de la phase de conception. Dans la littérature, divers travaux de recherche dans le domaine de gestion des processus métier (BPM) sont généralement axés sur la pers-

pective du flux de contrôle. Bien qu'il existe certaines approches axées sur la perspective des ressources, elles sont généralement dédiées à la perspective des ressources humaines. Ainsi, les travaux sur l'intégration de la perspective des ressources IoT dans les processus métier sont limités pour résoudre des problèmes liés à l'hétérogénéité. De même, dans le contexte des CPM, il n'existe aucune prise en charge de la configuration permettant de modéliser la variabilité des ressources IoT au niveau des CPM. Cette variabilité résulte des fonctionnalités spécifiques aux ressources IoT, telles que la possibilité de partage, et réplication, qui sont pertinentes dans le contexte des processus métier. Dans cette thèse, nous abordons les limitations susmentionnées en proposant une approche pour intégrer la perspective IoT dans le domaine du BPM et soutenir le développement de CPM. Ce travail propose les contributions suivantes : (1) il fournit une description formelle de la perspective des ressources IoT, et de ses relations avec le domaine BPM à l'aide de la technologie sémantique, et (2) il fournit de nouveaux concepts pour permettre l'allocation de ressources IoT configurables dans les CPM. Pour valider notre approche et démontrer sa faisabilité, nous procédons comme suit : (1) implémenter des outils preuve de concept qui soutiennent le développement de processus métier et de modèles de processus configurables conscient des IoT, et (2) réaliser des expérimentations sur des jeux de données de modèles de processus qui démontrent l'efficacité de notre approche et affirment sa faisabilité.

Title : Modeling the Internet of Things in Configurable Process Models

Keywords : Internet of Things ; Configurable Processes ; Modeling ; Business Process Management

Abstract : On the one hand, a growing number of multi-national organizations have embraced the Process-Aware Information Systems (PAIS) to reap the benefits of using streamlined processes that are based on predefined models, also called as Business Process (BP) models. However, today's dynamic business environment demands flexibility and systematic reuse of BPs, which is provided by the use of Configurable Process Models (CPMs). It avoids the development of processes from scratch, which is both time-consuming and error-prone, and facilitates the sharing of a family of BP variants that can be customized based on concrete business requirements. On the other hand, the adoption of the Internet of Things (IoT) resources in various cross-organizational BPs is also on a rise. However, to attain the desired business value, these IoT resources must be used efficiently. These IoT devices are heterogeneous due to their diverse properties and manufactures (proprietary standards), which leads to issues related to interoperability. Further, being resource-constrained, they need to be allocated (and consumed) keeping in the mind relevant constraints such as energy cost, computation cost, to avoid failures during the time of their consumption in the processes. Thus, it is essential to explicitly model the IoT resource perspective in the BP models during the process design phase. In the literature, various research works in Business Process Ma-

agement (BPM) domain are usually focused on the control-flow perspective. While there do exist some approaches that focus on the resource perspective, they are typically dedicated to the human resource perspective. Thus, there is limited work on integrating the IoT resource perspective into BPs, without any focus on solving issues related to heterogeneity in IoT domain. Likewise, in the context of CPMs, there is no configuration support to model IoT resource variability at the CPM level. This variability is a result of specific IoT resource features such as Shareability and Replication that is relevant in the context of BPs. In this thesis, we address the aforementioned limitations by proposing an approach to integrate IoT perspective in the BPM domain and to support the development of IoT-Aware CPMs. This work contributes in the following manner : (1) it provides a formal description of the IoT resource perspective and its relationships with the BPM domain using semantic technology and (2) it provides novel concepts to enable configurable IoT resource allocation in CPMs. To validate our approach and to show its feasibility, we do the following : (1) implement proof of concept tools that assist in the development of IoT-aware BPs and IoT-aware CPMs, and (2) perform experiments on the process model datasets. The experimentation results show the effectiveness of our approach and affirm its feasibility.



*To my family,
for their unconditional love.*

*To Aybüke, my loving wife and best friend,
for being an important part of this PhD journey.*

Acknowledgment

First, I thank God for giving me this opportunity and making this remarkable journey possible.

I would like to express my sincere gratitude to my thesis director Professor Walid Gaaloul for his patience and for being a wonderful mentor. I had numerous thought-provoking discussions with him, which have enriched me intellectually and helped me to develop a better understanding of research in general. I would like to thank my co-supervisor Professor Samir Tata for his valuable advice, guidance and encouragement. I thank my co-supervisor Dr. Arnaud Cuccuru for his gracious support, valuable feedback and providing a positive learning environment.

I would like to thank all the members of the jury. I thank Professor Chirine Ghedira Guegan and Professor Michael Mrissa for accepting to be my thesis reviewer and for their comprehensive and insightful comments. I also thank Dr. Maude Manouvrier, Dr. Wided Guédria and Professor Nazim Agoulmine for accepting to be my thesis examiner.

This PhD research project has been made possible with the support from many people belonging to both Telecom SudParis and CEA Paris-Saclay. Thus, I will like to express my gratitude to all of them for their help and especially to Brigitte and Frédérique. I would like to thank Dr. Juan Cadavid and Dr. Mauricio Alférez for their gracious support during the first year of my PhD thesis. I would also like to thank my fellow doctoral students and other colleagues from both CEA and Telecom SudParis for sharing their experiences, for having stimulating discussions and for all the fun talks during the last three years.

Finally, I would like to thank and dedicate this work to my family. I am eternally grateful to my parents, Sweety and Jagdish Chander Suri for their unconditional love and emotional support throughout my life. My success in life is a result of their relentless hard work. I would like to thank my wife Aybüke for her love and care. She has been my strength during the ups and downs of life. My warm thanks to my parents-in-law, Rabia and Kamil Öztürk for their love and kindness. I wholeheartedly thank my brother Karan for his love and for being a vital part of my life. I also thank my sister-in-law Aysena for her love and for spreading happiness in our lives. Last but not least, I am grateful to other members of our family and friends who have been there for us through thick and thin.

Thank you all once again!

Abstract

On the one hand, a growing number of multi-national organizations have embraced the Process-Aware Information Systems (PAIS) to reap the benefits of using streamlined processes that are based on predefined models, also called as Business Process (BP) models. However, today's dynamic business environment demands flexibility and systematic reuse of BPs, which is provided by the use of Configurable Process Models (CPMs). It avoids the development of processes from scratch, which is both time-consuming and error-prone, and facilitates the sharing of a family of BP variants that can be customized based on concrete business requirements. On the other hand, the adoption of the Internet of Things (IoT) resources in various cross-organizational BPs is also on a rise. However, to attain the desired business value, these IoT resources must be used efficiently. These IoT devices are heterogeneous due to their diverse properties and manufactures (proprietary standards), which leads to issues related to interoperability. Further, being resource-constrained, they need to be allocated (and consumed) keeping in the mind relevant constraints such as energy cost, computation cost, to avoid failures during the time of their consumption in the processes. Thus, it is essential to explicitly model the IoT resource perspective in the BP models during the process design phase.

In the literature, various research works in Business Process Management (BPM) domain are usually focused on the control-flow perspective. While there do exist some approaches that focus on the resource perspective, they are typically dedicated to the human resource perspective. Thus, there is limited work on integrating the IoT resource perspective into BPs, without any focus on solving issues related to heterogeneity in IoT domain. Likewise, in the context of CPMs, there is no configuration support to model IoT resource variability at the CPM level. This variability is a result of specific IoT resource features such as Shareability and Replication that is relevant in the context of BPs.

In this thesis, we address the aforementioned limitations by proposing an approach to integrate IoT perspective in the BPM domain and to support the development of IoT-Aware CPMs. This work contributes in the following manner: (1) it provides a formal description of the IoT resource perspective and its relationships with the BPM domain using semantic technology, and (2) it provides novel concepts to enable configurable IoT resource allocation in CPMs. To validate our approach and to show its feasibility, we do the following: (1) implement proof of concept tools that assist in the development of IoT-aware BPs and IoT-aware CPMs, and (2) perform experiments on the process model datasets. The experimentation results show the effectiveness of our approach and affirm its feasibility.

Table of contents

1	Introduction	17
1.1	Research Context	17
1.2	Research Problem: How to support the allocation of IoT resources in configurable business process models?	22
1.2.1	On Formalization and Modeling of IoT-Aware BPs in a Common Knowledge Base	23
1.2.2	On Supporting IoT Induced Variability Through Config- urable Resource Allocation	24
1.3	Motivating Example	24
1.4	Objectives and Contributions	30
1.4.1	Thesis Objectives	30
1.4.2	Thesis Contributions	31
1.5	Thesis Outline	32
2	Related Work	35
2.1	Introduction	36
2.2	Background	36
2.2.1	Process Modeling Languages	36
2.2.1.1	Business Process Model and Notation	37
2.2.1.2	Configurable BPMN	38
2.2.2	Basic Concepts from IoT Domain	41
2.2.3	Relevant Semantic Models from BPM Domain	43
2.2.3.1	Web Ontology Language for Web Services	44
2.2.3.2	Business Process Modeling Ontology	44
2.2.4	Relevant Semantic Models from IoT Domain	46
2.2.4.1	Internet of Things-Lite Ontology	46
2.2.4.2	Federated Interoperable Semantic IoT Testbeds and Applications Ontology	47
2.3	Support for the Development of IoT-Aware BP Models	48
2.3.1	Allocation of Resources in BPs	48
2.3.2	Allocation of IoT Resources in BPs	50
2.3.3	Synthesis	55

2.4	Semantic Technology in BPM and IoT Domain	56
2.4.1	Application of Semantic Technology in BPM Domain	57
2.4.2	Application of Semantic Technology in IoT Domain	59
2.4.3	Synthesis	60
2.5	Management of Resource Variability in CPMs	63
2.5.1	Support for Developing CPMs	63
2.5.2	Support for Managing Resource Variability in CPMs	66
2.5.3	Synthesis	67
2.6	Conclusion	68
3	Support for Modeling Semantically Enriched IoT-Aware Business Processes	71
3.1	Introduction	72
3.2	Extending the BPMN 2.0 with IoT Resource Perspective	73
3.2.1	Integrating the IoT Perspective in BPMN 2.0	74
3.2.1.1	IoT Specific Energy Perspective in BPMN 2.0	75
3.3	Semantic Formalization of IoT Perspective in BPs	77
3.3.1	Approach Overview	78
3.3.2	Ontology Concepts	79
3.3.2.1	IoT Services & Non-IoT Services	81
3.3.3	Ontology Attributes	84
3.3.4	IoT Resource Properties and Relationships in BPs	85
3.3.4.1	IoT Resource Properties in BPs	85
3.3.4.2	Relationships Between IoT and Other Resources in BPs	89
3.3.5	Managing IoT Resource Conflicts using Strategies	92
3.3.5.1	Constraints on IoT Resources and Task in BPs	92
3.3.5.2	Strategies for Resolving IoT Resource Conflicts	93
3.4	Evaluation and Validation	96
3.4.1	Ontology Validation	96
3.4.2	Support for Developing IoT-Aware BP Models	99
3.4.2.1	Integrating IoT Resource Definitions in BP Models	99
3.4.2.2	Semantic Annotations in BP Models	99
3.4.3	Ontology-based Knowledge Base	100

Table of contents	11
3.4.3.1 Querying the Knowledge Base	101
3.4.4 Threats to Validity	102
3.5 Conclusion	103
4 Supporting IoT Induced Variability in CPMs	105
4.1 Introduction	105
4.2 IoT Resource Perspective in BPs	107
4.3 Configurable IoT-Aware Allocation in CPMs	109
4.3.1 Configurable IoT Assignment Operator	110
4.3.2 Configurable IoT Shareability Operator	112
4.3.3 Configurable IoT Replication Operator	114
4.4 Developing IoT-Aware CPMs: Our Approach vs Classical Approach	116
4.5 Evaluation and Validation	117
4.5.1 Supporting Configurable IoT-Aware Process Modeling . . .	118
4.5.2 Experimentation	119
4.5.3 Threats to Validity	121
4.6 Conclusion	122
5 Conclusion and Future Works	125
5.1 Conclusion	125
5.2 Future Works	127
Appendices	129
A List of Publications	131
Bibliography	133

List of Tables

1.1	IoT resource variability in process variants	30
2.1	Constraints regarding the configuration possibilities of the gateways	40
2.2	Existing approaches with resource perspective in BPM	57
2.3	Existing semantic models for BPM and IoT domain	61
2.4	Existing semantic models relevant to our work	62
2.5	Evaluation of existing approaches for developing CPM	68
3.1	Summary of IoT specific properties for developing IoT-aware BPs .	88
3.2	Relationships between IoT and other resources	91
3.3	Strategies for conflict resolution using SWRL formalism	95
3.4	Coverage of concepts in IoT-A reference	97
3.5	Existing semantic models compared to our approach	98
4.1	Parameters for configurable IoT assignment operator	112
4.2	Parameters for configurable IoT shareability operator	113
4.3	Parameters for configurable IoT replication operator	116
4.4	Complexity metrics comparing different approaches	121

List of Figures

1.1	Business Process Management lifecycle as per [1,2]	19
1.2	Configuration and resource allocation in the BPM lifecycle	21
1.3	A supply chain process model illustrating the monitoring of temperature of a physical object	26
1.4	Configuration and individualization of a CPM	27
1.5	CPM from Supply Chain Management domain	28
1.6	Process Variant-1 derived from Figure 1.5 based on control-flow perspective	29
1.7	Process Variant-1 in Figure 1.6 enriched with IoT resource features	29
2.1	Graphical representation of BPMN elements	38
2.2	Process models having common and variable parts	39
2.3	Configuration of activity and gateways	40
2.4	Types of devices in the IoT domain	42
2.5	Example of various types of network for connecting IoT devices . .	43
2.6	The process ontology in OWL-S [3]	45
2.7	High-level overview depicting IoT-BPM interaction [4]	51
2.8	Functional view of the IoT Architectural Reference Model [5] . . .	52
2.9	BPMN 2.0 extension with IoT concepts [6]	52
2.10	IoT concepts relevant for BPM domain [6]	53
2.11	IoT architectural layer with WfMS layer [7]	55
2.12	An example representing a C-EPC process model [8]	64
2.13	Configuration from OR to AND by hiding and blocking [9]	65
2.14	Provop process variant lifecycle [10]	66
3.1	Extending resource element in BPMN for IoT devices	76
3.2	Overview of the approach	79
3.3	Concepts in ThingsPrO ontology	83
3.4	Lifecycle depicting states of a task instance	84
3.5	Lifecycle depicting states of an IoT resource	84
3.6	Modeling IoT resource and ontology annotation in BPMN 2.0 . . .	100
3.7	Screenshot depicting information retrieval from KB	102

4.1	Process fragment taken from Figure 1.5 to illustrate configurable IoT allocation operators	110
4.2	Comparison of approaches to develop IoT-aware CPMs	117
4.3	Screenshot illustrating the implemented proof of concept	118
4.4	Process fragments illustrating three different approaches	122

Introduction

Contents

1.1	Research Context	17
1.2	Research Problem: How to support the allocation of IoT resources in configurable business process models?	22
1.2.1	On Formalization and Modeling of IoT-Aware BPs in a Common Knowledge Base	23
1.2.2	On Supporting IoT Induced Variability Through Configurable Resource Allocation	24
1.3	Motivating Example	24
1.4	Objectives and Contributions	30
1.4.1	Thesis Objectives	30
1.4.2	Thesis Contributions	31
1.5	Thesis Outline	32

1.1 Research Context

In the modern economy, organizations need to be more efficient and cost-effective to survive the ever-increasing competition. In other words, these organizations must develop, execute and manage their complex processes (such as order processing, purchasing, production and logistics or financial processes) keeping in mind the dynamic markets. They should also keep in mind the continuous changes that they face every now and then such as new customer needs or new government policies. Thus, it is crucial for the organizations to imbibe technologies and underlying information systems (IS) that support their need for flexibility and reuse. In this context, *Process-Aware Information Systems* (PAIS) [11–13] have emerged as a promising solution to enable efficient “process-oriented” management and execution of complex processes involving both systems and people on the basis of specific process models [12]. Most typical examples of these PAIS are the Workflow Management Systems (WfMSs) [14, 15] and the Business Process Management Systems (BPMSs) [12, 16, 17]. In fact, several real cases from the industry [18] illustrate and ensure the advantages of imbibing these PAIS, which

has helped different organizations such as Lufthansa [19], Siemens [20], Deutsche Bahn [21], Zalando SE [22], to name just a few, to achieve the desired business transformation so as to remain effective in today's dynamic business environment.

Business Process Management (BPM) is a field in operations management that is focused on improving the performance of an organization and the overall value generated by them through the optimization of their business processes (BPs). A process model also referred as a *Business Process Model* is the key component in BPM. It consists of various steps, i.e., activities (or tasks) and their execution order along with certain perspectives (behavioral or organizational) taking place in an organization over a range of time and at various locations. In literature, various process-modeling languages have been proposed, each having its own specific graphical representation but with the same underlying essence. The most notable examples of these modeling languages are: Business Process Model and Notation (BPMN) [23], Event-driven Process Chain (EPC) [24], XML Process Definition Language (XPDL) [25], Petri Nets [26] and Unified Modeling Language (UML) Activity Diagram [27, 28]. Furthermore, to support continuous improvement of the BPs, the BPM lifecycle is mainly categorized into four phases that are, *Process Design*, *Process Implementation*, *Process Execution* and *Process Diagnoses* [2, 11] (see Figure 1.1). In fact, the process design phase is the initial and a crucial phase in the BPM lifecycle [1]. This is because the errors introduced in the design phase will propagate to other phases resulting in wastage of effort, time and resources, and finally requiring a re-design of the process model itself. Thus, it is critical to properly design the BP models based on the specified business requirements and to analyze them (i.e., validation and verification) along with performing simulations on them to check for the desired outcomes. Likewise, in the process implementation phase, various tools and techniques are used to automate the BPs into executable processes. In the third phase, the BPs are executed by deploying these processes on a PAIS. In the final phase, i.e., process diagnose phase, the executions of the process are analyzed (using logs and traces) to check the process deviation and possible bottlenecks. This phase helps in redesigning and improvement of these processes.

Today, various complex BPs having a “physical character” (i.e., interaction with the physical world) are executed either by a single organization or collaboratively by a set of autonomous organizations [29]. Such processes can be found in several business domains such as supply chain and logistics (Industry 4.0 [30]), healthcare, smart home automation, to name just a few. For example, in case of an integrated supply chain and manufacturing networks, a network of companies providing transportation and administration services collaborate to enable the process of physically moving a container(s) (containing goods) from one geographical location to another. Both the containers and their contents (perishable or non-perishable goods) are the physical items to be managed along with the

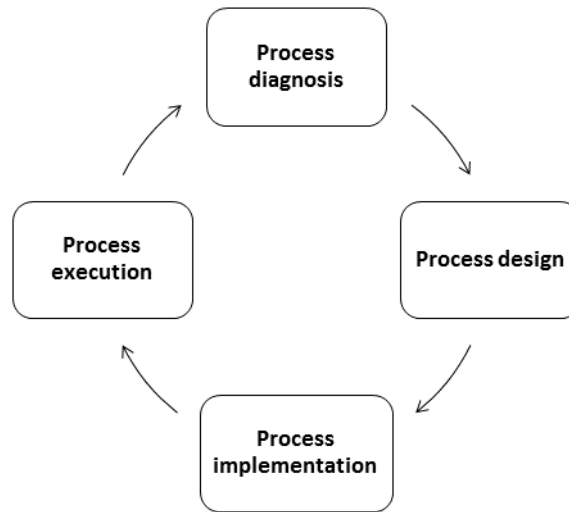


Figure 1.1: Business Process Management lifecycle as per [1,2]

management of resources such as the vehicles (trucks, ships) and the robots that support the overall process [29]. These processes rely heavily on the use of various heterogeneous devices (and their data) connected over the internet, which need to be orchestrated in a specific sequence to achieve the desired outcome. These connected devices form the Internet of Things (IoT) and based on their granularity can be broadly classified into *Sensors*, *Actuators*, and *Tags* (such as Radio Frequency Identification (RFID) tags) or simply into “things” such as robots (i.e., a specific combination of sensors, actuators and computation unit) and smart objects (i.e., object with embedded sensors, actuators or tags). They enable sensing, actuating (or reacting) and exchanging or collection of data through a communicating network such as the internet [31]. Traditionally, in the BPM domain, the information about the event and processes occurring in the physical world are linked to the digital world via the data entered by the humans or via web services [32–34]. These humans also assist in controlling the outcome of the physical world based on the data coming from the processes. However, with the use of IoT devices both the information about the physical world and the control over the physical world can be achieved instantaneously through the information systems. In fact, many such organizations rely heavily on the use of PAIS such as BPMS [35] to efficiently manage their processes [22]. Nonetheless, to optimally manage their allocated resources, i.e., both human and non-human (devices and systems), these PAIS need to become resource-aware [36] and further evolve into *Process- and IoT-Aware Information Systems*. Thus, it is important to effectively model and manage the IoT resource allocations in the BPs so as to effectively orchestrate these IoT resources in BPs.

Despite growth in research on the integration of BPM domain (and underlying technologies) with IoT domain [29, 37–39], still there exists several gaps and research challenges to foster the optimal allocation and management of IoT resources in BPs [4]. In literature, the research work in the area of BPM has been more focused around the control-flow perspective with some works focusing on the resource perspective in BPs for the management of human resources [36, 40–44]. However, there has been a lack of work done towards tackling the specificity related to the IoT domain such as heterogeneity in BPs. In BPM domain several research work [45–56] have also proposed the use of semantic technology (Semantic Web) to enrich the process models with semantic annotations. These semantic approaches in BPM domain help to solve the problems related to heterogeneity in BP models due to the use of various modeling languages in an organization. They also enable the application of formal reasoning techniques in order to assist in discovery, composition, mediation, and execution of BPs [57]. Such semantic technologies can also assist in formalizing the concepts and complex relationships from the IoT domain in the context of BPM domain. Likewise, several works [5, 58–61] have applied the semantic technology to the IoT domain to tackle the problems related to heterogeneity of devices and proprietary data formats so as to promote interoperability. However, to the best of our knowledge, there has been no uptake on realizing a semantic integrating of the two domains.

On another side, these complex and collaborative processes incorporating IoT resources have to withstand the dynamic markets and other situations such as rapidly changing business requirements, customer needs or government regulations in the context of smart ecosystems. This forces these organizations to imbibe PAIS that support flexibility and reuse of knowledge. In other words, these systems must facilitate the “Principle of Reuse” for modeling and/or (re-)designing the processes by taking into consideration the preexisting knowledge about similar processes and/or best practices existing in an organization, rather than forcing analysts to design processes “from scratch”. Such flexibility and reusability for modeling BPs are backed by the use of Configurable Process Models (CPMs) [8], which is an active area of research for managing process variability in BPM domain [62]. A CPM consolidates various process variants (multiple process solutions) into one customizable process model via variation points called *configurable elements* (activity or gateways) [63] (see Figure 1.4). In other words, a consolidated customizable model captures a family of process variants. This helps to avoid redundancy and allows improvements efforts made on one BP variant to benefit other variants. The classical approaches in CPM focus mainly on configuring the *control-flow perspective* [62], without giving much consideration to the resource perspective. Even though a few a limited proposals consider the extension of configuration to resources [10, 64, 65], they are too generic to tackle the complexity and specificity involved in the IoT domain (i.e., IoT specific features

(properties), constraints, and deployment strategies). Thus, this thesis focuses on formalizing and integrating the IoT perspective to BPs along with semantic enrichment of the BP models. Then, these IoT concepts are included in the CPM level to support the variability management of IoT-aware CPMs.

Overall, Figure 1.2 (adapted from Figure 1.1) illustrates the scope of this thesis, which is focused on the process design phase of the BPM lifecycle [1]. As our work is focused on developing IoT-aware CPMs, the process design phase of the BPM lifecycle in Figure 1.1 is updated and replaced with the “Configurable Process” design phase. Next, this phase is enriched with the IoT resource perspective such as selection and assignment (i.e., IoT resource allocation). Moreover, the configuration process design phase comprises of the process configuration and individualization step. On the whole, these three steps are realized at the design-time and to be more precise, the process configuration and individualization step fall into the configuration-time, which is a subset of the design-time.

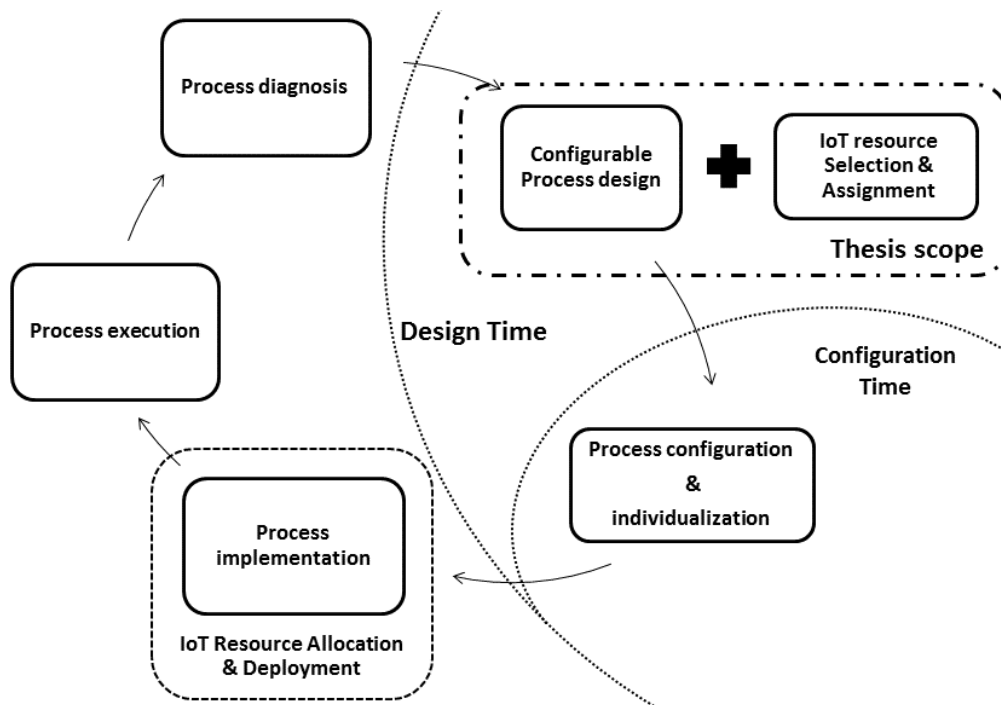


Figure 1.2: Configuration and resource allocation in the BPM lifecycle

1.2 Research Problem: How to support the allocation of IoT resources in configurable business process models?

Various research initiative in the BPM domain gives considerable emphasis to the effective management and orchestration of resource (mainly human and some work on systems) involved in the BPs. This is because the resource orchestration, which includes steps such as resource selection, allocation, deployment, monitoring, and control, is vital for the optimal execution of the processes. Especially because these processes may involve resources that are being used by different autonomic organization collaborating together, spread over different geographies and time zones, having specific geopolitical concerns such as privacy and security. Thus, making the effective use of resources becomes a top priority. In some cases, BPs involve both resources, i.e., humans and devices that are costly and scarce such as processes in healthcare. In the scope of this thesis, our work is focused on the modeling of the allocation behavior (and integration) of the IoT resources that will be consumed in the BPs.

On the one hand, numerous organizations in various domains such as supply chain, healthcare, smart home automation, to name just a few, make use of IoT devices in their BPs (whether configurable or not). They need to consume these IoT resources that are manufactured by different companies having their own proprietary standards. To solve the issue arising from the heterogeneity of IoT resources, various research initiatives have focused on standardization of IoT resources but still, we are far from having a unified standard for the IoT devices. Thus, it becomes crucial to have a unified understanding of the concepts and complex relationships between the domain of IoT and BPs (see Section 1.2.1). On the other hand, organizations competing in the dynamic markets should be able to reuse their preexisting knowledge about their processes including the information about the IoT resources being consumed. This may enable them to be flexible and adapt quickly to changes. It may also ensure a uniform adoption of the changes throughout the various locations of their organization. Moreover, the diversity and heterogeneity of the IoT resources (devices and networks) in the IoT domain leads to an increase in the variability of BP models. This calls for a need to support the configuration of IoT resources at the CPM level so that the process analysts can configure the IoT resource perspective with the same convenience such as the control-flow perspective, which has been a well-studied topic in the literature (detailed in Section 1.2.2).

1.2.1 On Formalization and Modeling of IoT-Aware BPs in a Common Knowledge Base

In the recent years, the research towards the integration of the IoT and BPM domains has generated interest because of the increase in the incorporation of the “physical character” (i.e., interaction with the physical world) in various complex BPs [29, 37, 38, 66]. These BPs use IoT devices (and data) during the execution of these processes. Nonetheless, the work related to the development of IoT-aware BPs is still in the nascent stage [4]. In general, due to the scarcity and cost-related issues of resources, several existing works in the BPM domain have focused on the efficient management of the resource perspective in BPs. However, most of these works have focused on the integration of human resource perspective in BPs so as to use the human workforce in a better manner for cognition intensive tasks [67]. Moreover, the IoT domain needs special attention due to a high level of heterogeneity in the IoT resources (devices, networks) and the lack of standardization (disparate manufacturers with proprietary standards). This calls for developing a uniform, formal definition for various IoT concepts that will foster the interoperability between different IoT resources from different providers. Despite some existing work on the formalization of the concepts and semantics of IoT domain [5, 58–61], there has been no uptake to semantically integrate these concepts with the BPM domain.

This thesis work proposes to bridge the above-mentioned research gap by doing the following: (1) extending the resource perspective to include IoT resources in BPMN for modeling IoT-aware BPs, (2) developing a unified cross-domain semantic model that integrates the IoT concepts with BP concepts along with their complex relationships. This semantic model is developed by considering the best practices from the ontology-engineering domain, i.e., reusing concepts from existing semantic models. It provides a framework for a correct selection (and assignment) of these IoT resources and the possibility to manage their allocations in a conflict-free manner (during BP execution). Overall, to address our research problem, the following sub-questions need to be answered:

- RQ1: How to uniquely integrate the IoT resource perspective in BP models?
- RQ2: How to formally define and include the specific IoT concepts along with their relationships into BPs?
- RQ3: How to formally define heterogeneous IoT resources and their allocations along with the semantically enriched BP models to be shared in a common knowledge base?

1.2.2 On Supporting IoT Induced Variability Through Configurable Resource Allocation

The dynamic modern economy with its ever-changing business requirements, customer needs or government regulations have forced the organizations to find technical solutions that may help them to become flexible and support reuse of the already existing process knowledge, while keeping it all cost-effective. In the context of the process-oriented management technology (such as BPM), the CPMs provide the needed flexibility for such a dynamic environment as it is based on sound foundations of a process-oriented approach, which works on a predefined sequence of steps or tasks. This enables these organizations to be ready for changes (by reusing process knowledge) while making sure that already running businesses execute with confidence, their performance is up to the mark and with low maintenance costs.

In other words, such a dynamic business environment calls for the need of flexibility (in a process-oriented way) and management of variability that is supported via CPMs [8]. CPMs facilitate reuse in process (re-)design by considering the preexisting knowledge about similar processes. In literature, several classical approaches exist for modeling CPMs, wherein most of them are focused on the control-flow perspective. Despite some initial work on the resource perspective in CPMs, there is no work that integrates IoT specific features at the CPM level. Thus, the current research area on CPM lacks approaches to manage the variability induced by the IoT specific features and its behaviors. These features are introduced due to the specific requirements of BPs based on the need for privacy (shareability) and availability of the IoT resources (replication). To address this research problem, the following questions need to be answered:

- RQ4: How to integrate the variability induced by IoT resources and their features at the CPM level?
- RQ5: How to assist the process designers to configure their choices with respect to IoT resources?

1.3 Motivating Example

In the real-world, many BPs imbibe certain physical characteristics. Such processes can be found in several domains such as logistics, healthcare, smart homes, to name just a few [29]. We motivate this thesis work using examples from the supply chain and logistics domain. To keep the examples sound, these process models are adapted from the literature [66, 68], which were developed and used in the EU FP7 project, *Internet of Things Architecture* (IoT-A¹). These exam-

¹ IoT-A project: http://cordis.europa.eu/project/rcn/95713_en.html

ples have been modeled in BPMN [69] but they can be easily extended to other modeling languages such as EPC. The examples detailed in this section are used as running examples to explain the contributions in the following chapters of the thesis.

Figure 1.3 represents a BP for monitoring the condition of goods in a supermarket or in a warehouse. An adaptation of the same monitoring process is applied to a container (logistics) transporting goods from one location to another via a truck or ship. Figure 1.3 represents a self-triggering process (every 60 seconds) wherein an activity *a1* is used to measure the temperature of a physical object such as a Chinese Orchid flower. This temperature data is sent to the *Backend Application* via activity *a2*. While the temperature is within a pre-described range, the process ends without any alert. Otherwise, the activity *a3* raises an alarm message. The activity *a4* is used to estimate the degradation in the quality of the item based on some pre-described algorithm (out of the scope of this work) and this information is stored in a *Sensor Historical Datastore*. If the estimated quality is within a pre-described range, the process ends without an alert. Otherwise, with the use of activity *a5*, it checks if there was a temperature alarm message and then reduces the price of the physical entity, i.e., Orchid. The activity *a5* sends a message to both: (1) the *Cashier System* that updates the price of the physical entity in the system via activity *a6*, and (2) the *Electronic Shelf*, which updates the price of the product via activity *a7* based on the message from activity *a5*. This monitoring process enables a supply chain to serve its customers in a better way by following a dynamic pricing mechanism based on real-time information from the sensor.

The BP example in Figure 1.3 will need to integrate (“glues”) several IT systems and IoT devices together. Thus, before the actual implementation of this process takes place, all the underlying technical details related to the IoT devices (and other systems) must be included in the BP models. During the actual deployment of the process depicted in Figure 1.3, the BP involves the use of information from IoT devices such as a sensor and RFID device to complete the tasks. The activity *a1* will be associated with a sensor (for instance *sensor1*), while the activity *a7* will be associated with an RFID Tag (for instance *RFID1*). Additionally, it is to be noted that these IoT resources participating in the process have specific features such as energy consumption cost, network usage cost and specific behaviors such as privacy (and shareability). For instance, an RFID resource can be associated with a single activity in a single process or multiple activities in different processes. While a sensor can be associated with a single activity, or it can share its data via a publish-subscribe middleware. However, the current state-of-the-art in BPM domain does not support such resource allocation that considers the integration of relevant IoT features. In addition, there is no support for a formal description of concepts from the IoT domain and their relationships

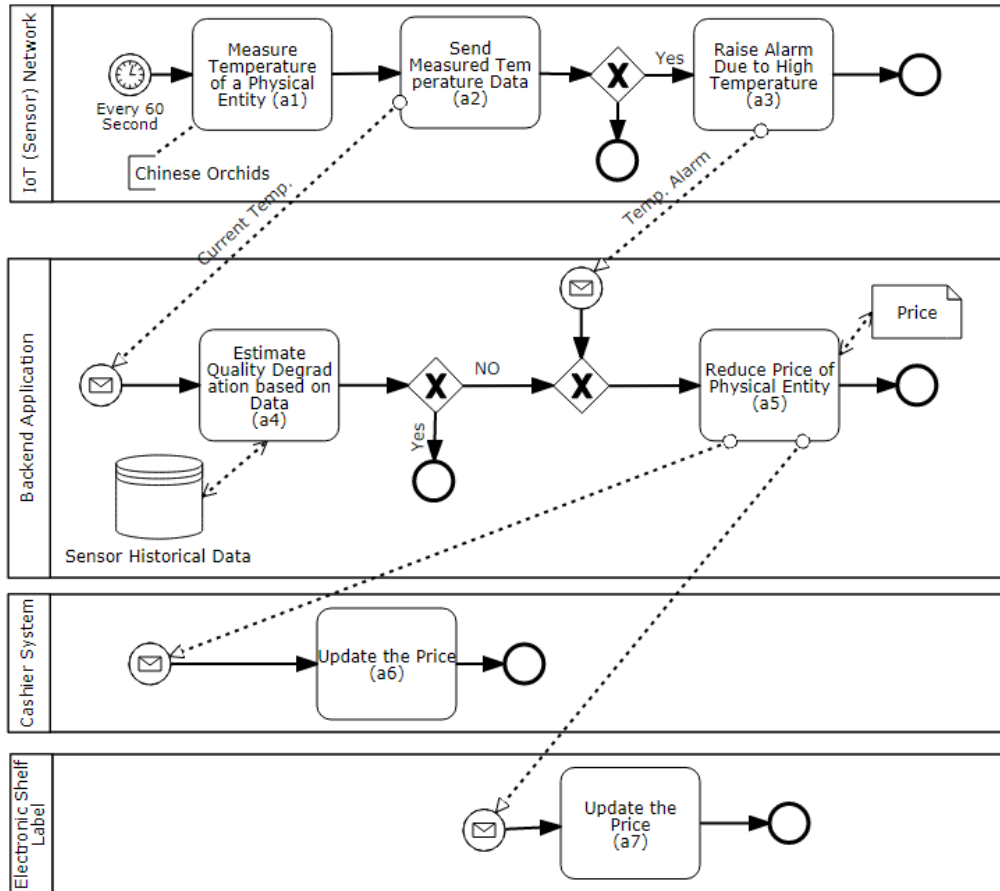


Figure 1.3: A supply chain process model illustrating the monitoring of temperature of a physical object

with concepts of BPs.

Furthermore, in order to achieve flexibility and to reuse the existing process knowledge an organization makes use of CPMs as represented in Figure 1.5. Nonetheless, before diving into the example of the CPM in Figure 1.5, we need to understand some basic notions of CPMs. Through Figure 1.4, we briefly illustrate these underlying concepts from CPMs such as configuration and individualization (detailed in Section 2.2.1.2). Figure 1.4 depicts a CPM modeled using Configurable BPMN (C-BPMN), which is an extension of classical BPMN (see Section 2.2.1). Similar to BPMN, the classical C-BPMN also has control-flow elements but they are configurable and graphically represented via thick lines. The CPM in Figure 1.4 contains five activities: A_1 , A_2 , A_3 , A_4 and A_5 . The activity A_2 , A_3 are connected via a configurable OR gateway (OR^C) shown with a thick line. The activity A_3 being a configurable activity is depicted via thick lines. The point to note is that unlike ordinary gateways (e.g. the AND gateway between A_4 and A_5) the configurable gateways do not represent a *run-time decision*. Rather,

they represent a design-time choice that shall be made by a business analyst for developing process variants (based on business needs). For instance, one analyst needs to configure the CPM into a variant without having the $A3$ activity (Process configuration-1 in Figure 1.4. While another analyst needs to keep $A3$ and to adapt the Configurable OR into an XOR gateway. These choices are visible in the Process variants 1 and 2. As evident from the literature, the classical concepts in CPMs do not consider the inclusion of IoT specific features in the CPM models and thus they do not support the management of variability due to IoT resources.

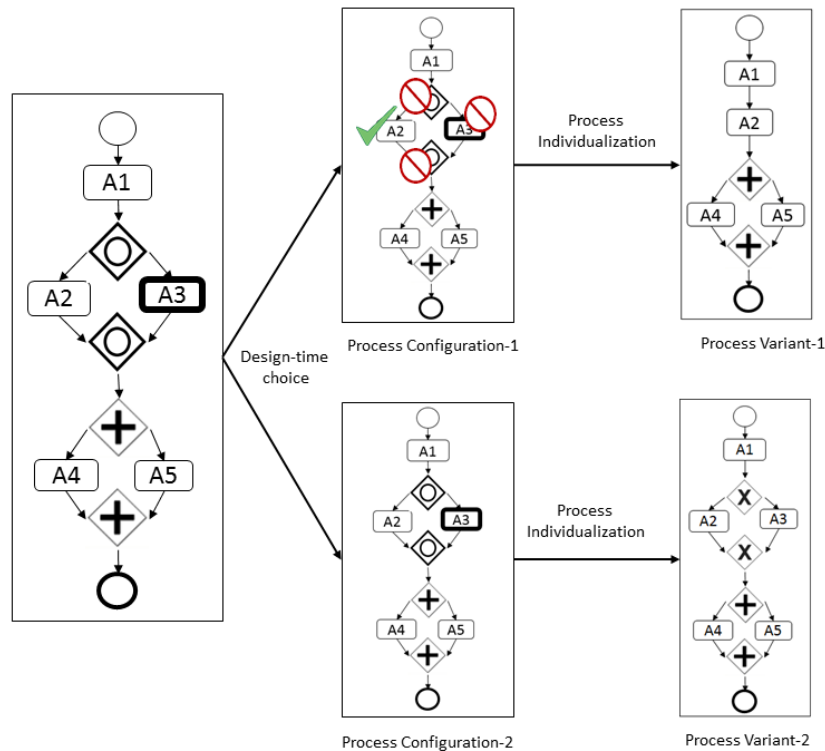


Figure 1.4: Configuration and individualization of a CPM

To illustrate the use of CPMs in organizations, in Figure 1.5 we present a CPM that is an extension of the process in Figure 1.3. In other words, based on the business needs an analyst can customize the CPM in Figure 1.5 to derive the BP model in Figure 1.3. This CPM is developed using c-BPMN and it represents the consolidated view for a collection of BP model for monitoring a supply chain [68] based on algorithms presented in [70, 71]. It will assist retailers to share their process knowledge and policies (rules and constraints) in a reusable and customizable manner with their affiliates spread across the globe. The process starts based on a timer event for enabling periodic monitoring of goods. After the process starts, there are two possibilities represented by two sub-processes interconnected via a configurable XOR gateway (for instance $XORc-1$). The sub-process I supports the monitoring of an item from *fast-moving consumer goods* (FMCG) category such

as vegetables, cheese, flowers, while the sub-process II supports the monitoring of durable goods such as TV, shoes, to name just a few.

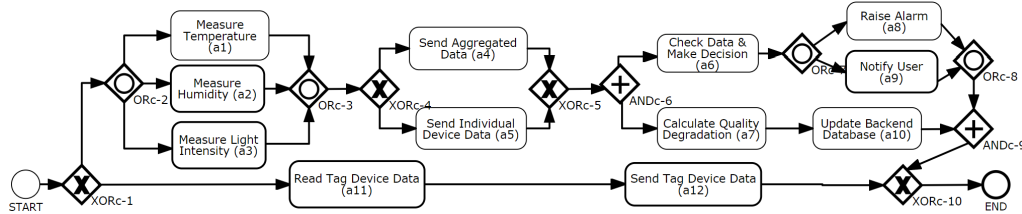


Figure 1.5: CPM from Supply Chain Management domain

Next, to demonstrate the need of including IoT-resource perspective at the CPM level, we first describe a process variant, i.e., *Variant-1* (represented in Figure 1.6), which is derived from the above mentioned CPM (see Figure 1.5) based only on the control-flow perspective. The process in Figure 1.6 is a simplified version of the process in Figure 1.3. Next, we show how including the IoT resource perspective increases the complexity of this process variant. Let us assume that a French retailer such as Carrefour at a location A (say Paris), decides to individualize the CPM to include only a temperature monitoring step for a perishable item such as Chinese Orchids flower, i.e., similar to BP in Figure 1.3. Thus, at the design-time an expert will customize the CPM (modeled in C-BPMN) into a process variant, i.e., Variant-1 (modeled in BPMN). The Variant-1 is configured to include activities $a1$, $a5$, $a6$, $a7$, $a8$, $a10$ (see Figure 1.6). The derivation (individualization) of a process variant based on the classical control-flow perspective is done by removing the unwanted nodes. Nonetheless, for efficient resource management, there is a need to explicitly capture the knowledge about the IoT resources (i.e., IoT properties, behavior and deployment strategies), which should also be included in the BP model. For instance, based on some business needs, the activity $a1$ needs a *digital temperature sensor* having high-accuracy, i.e., accuracy of $\pm 0.5^\circ\text{C}$ (max) from 0°C to $+65^\circ\text{C}$ (e.g., a TMP112² sensor from Texas Instruments (TI)). Additionally, during deployment, this device will need a network resource, i.e., *Network-01*, which should be long range, consumes lower power and allows secure data transmission such as Low Power Wide Area Network (LPWAN) based LoRaWAN³. Further, this resource can be deployed on a public cloud infrastructure. All these parameters and information depict the IoT specific features, i.e., *Resource Properties*, which should be included in the process models. This information is needed to support the implementation and deployment of the IoT-aware BP during the next phases of the BPM lifecycle [1] (out of the scope of this work).

Figure 1.7 represents a BP that reuses the Variant-1 (see Figure 1.6) and

²TI's TMP112 - <http://www.ti.com/lit/ds/symlink/tmp112.pdf>

³<https://www.lora-alliance.org>

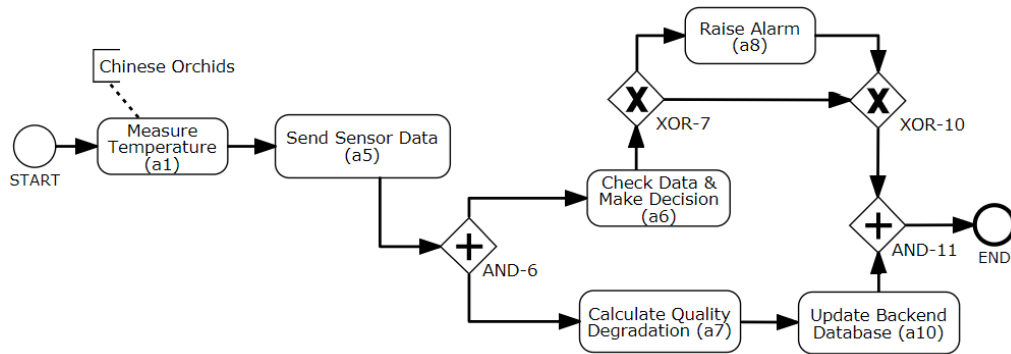


Figure 1.6: Process Variant-1 derived from Figure 1.5 based on control-flow perspective

enriches it with information about the IoT resource features in form of text annotations. The IoT resource consists of specific *Resource Behavior* that should be included in the process models. For instance, a device and the network can be Shareable, i.e., it shall share its data using publish/subscribe (pub-sub) middleware. Additionally, the activity $a1$ can be connected to more than one temperature sensor provided they exhibit similar capability, i.e., aggregation of a set of similar physical devices via a logical interface. This results in improvement of availability, fault-tolerance, and helps to achieve higher Quality of Information (QoI) [39] (see Section 3.3.4). These resource behaviors are also included as text annotation as observable in the Figure 1.7.

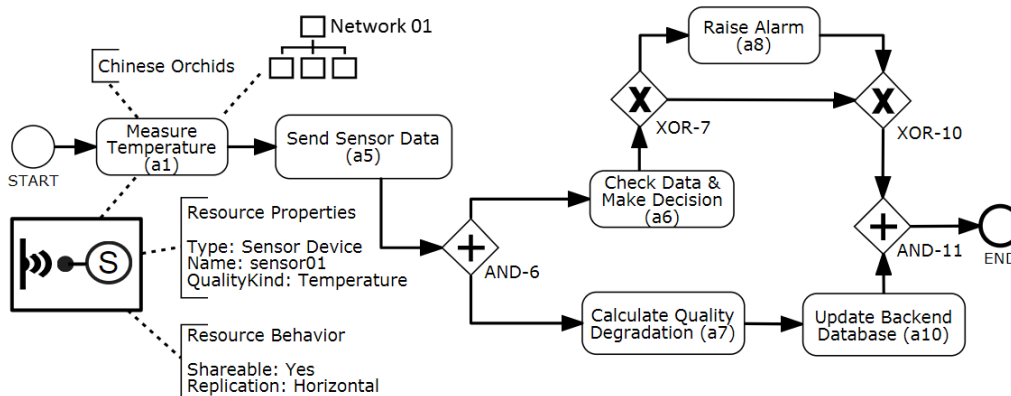


Figure 1.7: Process Variant-1 in Figure 1.6 enriched with IoT resource features

Now, let's say another Carrefour market at a location B (say Brussels), decides to individualize the CPM in Figure 1.5 into another BP variant, i.e., *Variant-2* (not represented as a figure). Based on the business needs the BP Variant-2 has the same control-flow as that of the BP Variant-1, but different IoT specific requirements. For instance, in the BP Variant-2, activity $a1$ requires a low-accuracy

digital temperature sensor with Accuracy of $\pm 2^\circ\text{C}$ (max) from -40°C to $+125^\circ\text{C}$ (e.g., TI’s TMP103) and a cellular network resource (*Network-02*). Similarly, there could be another BP variant, i.e., *Variant-3*, having same control-flow as BP Variant-1 but requiring a *low-power dust resistant* sensor (e.g., TI’s HDC1080). Additionally, this resource can be deployed using both cellular network or LoRa network depending on the availability at the deployment time. Table 1.1 illustrates the complexity involved in capturing the IoT resource variability while considering just a single activity (*a1*) from the CPM.

Variant	Control-Flow	Resources	Resource Property	Resource Behavior
<i>Variant-1</i>	Derived from CPM	Sensor, Network	High-Accuracy (HA)	Shareable
<i>Variant-2</i>	Same as <i>Variant-1</i>	Sensor, Network	Low-Accuracy (LA)	Shareable
<i>Variant-3</i>	Same as <i>Variant-1</i>	Sensor, Network	HA & Low-Power	Non-Shareable

Table 1.1: IoT resource variability in process variants

These examples clearly illustrate that the process variants share commonalities not only at the *structural* and *behavioral* level (i.e., control-flow perspective) but even at the resource level. In practice, various variants have similar requirements for the allocated resources with slight changes such as choice of accuracy, network, capability, deployment strategies, or shareability (i.e., resource behavior). However, not having a configuration support to model this resource variability at CPM level, causes several disadvantages: (1) the allocation parameters are hard-coded at each individual variant level in an ad hoc manner, (2) there is no knowledge coming from CPM level, i.e., no guidance (rules or constraints), (3) variant creation is time-consuming and error-prone, (4) the process enrichment (and best practices) takes place at the variant level of the BP models, leading to redundancy and segregation of improvement efforts for each variant, without benefiting others. Thus, we advocate shifting the IoT resource allocation behavior from the BP variant level to the CPM level.

Overall, in this thesis we propose to do the following: (1) the integration of the formal definition of IoT concepts into the BPM domain, (2) support for modeling the IoT resource allocation behavior along with verification of the constraints such as energy cost, and (3) support for including the IoT resource variability at CPM level via configurable operators.

1.4 Objectives and Contributions

1.4.1 Thesis Objectives

The aim of this thesis is to provide support for integrating the IoT perspective into CPMs. The core objectives of this work are as follows:

- **Objective-1:** *to enable IoT resource allocation in BPs by integrating the formal definitions of the IoT resource perspective into BP models.* This objective is achieved by the following: (1) using both extending the modeling capability of BPMN with IoT resource perspective, and (2) by using semantic technologies. It supports the formalization of IoT concepts with BPM concepts through the development of a unified cross-domain semantic model. This semantic model provides a framework for ensuring correct usage of IoT resources and helps to avoid conflicts due to resources. It is also used to generate a machine-readable ontology-based knowledge base, which can be used for reasoning purposes during the execution time.
- **Objective-2:** *to manage the variability induced by IoT resources in CPMs.* This objective is realized through the proposal of a novel approach that introduces configurable IoT resource allocation operators. These operators allow the modeling of resource alternatives at the CPM level that are induced due to the IoT specific features.

1.4.2 Thesis Contributions

In this thesis, the above-mentioned objectives are achieved by the following two main contributions:

The first contribution proposes a semantic framework for formalizing the IoT resources definition in the context of the BPM into a common knowledge base. In this approach, the BPs are semantically enriched with information related to IoT concepts and specifics. This semantic framework for developing IoT-aware BPs supports the following: (1) a unique and formal manner to define the concepts and relationships of IoT resources with BP concepts, (2) management of IoT resource allocation, (3) verification of constraints and conflict avoidance strategies for IoT resources. Overall, this contribution comprises the following steps:

- Extension of the resource concept in the BPMN 2.0 meta-model to include the IoT resource perspective,
- Development of a unified cross-domain semantic model called ThingsPrO ontology, which integrates the concepts from the BPM domain (using BP MO ontology [48,72,73]) with the concepts from the IoT domain (using FIESTA-IoT⁴ ontology) along with formalizing relationships and dependencies between these two domains,
- Creation of constraints and rules to assist the resolution of resource-based conflicts,

⁴<http://fiesta-iot.eu/>

- Creation of a knowledge base with heterogeneous IoT resources and BP models,
- Development of a proof of concept tool to allow the modeling of BPs with IoT resources and semantic enrichment of these models with the concepts from the ThingsPrO ontology.

The second contribution aims to provide a novel approach to enable modeling of configurable IoT resource allocation based on the two main properties, i.e., shareability and replication. This approach provides allocation operators that assist in shifting the allocation of IoT resources from individual process level to the level of CPMs. Such a shift assists the process designers by reusing the knowledge from the CPMs and avoiding duplication of effort at the individual process level. Overall, this contribution has the following steps:

- Development of the new configurable IoT resource allocation operators based on the IoT properties of shareability and replication,
- Conducting experiments on process datasets to demonstrate the feasibility and effectiveness of our approach,
- Development of a proof of concept to enable the modeling of the configurable IoT resource allocations in the CPMs.

1.5 Thesis Outline

This thesis is divided into five chapters that are organized as follows:

- **Chapter 2: Related Work** reviews the existing solutions in the literature with respect to the research problems detailed in this thesis. It helps to position our contributions with the existing work. In this chapter, we present the existing approaches on the modeling of IoT resource perspective in BP models. We also present the solutions for modeling CPMs and the extensions of CPMs that consider the resource perspective.
- **Chapter 3: Support for Modeling Semantically Enriched IoT-Aware Business Processes** describes our approach to model IoT-aware BPs and semantically enrich them using the cross-domain ontology developed in this work. This ontology reuses the concepts from existing ontologies, namely BPMO and FIESTA-IoT ontology. The semantic model is used to populate a common knowledge base of IoT resources that support the design of IoT-aware BP models.

- **Chapter 4: Supporting IoT Induced Variability in CPMs** details our approach to support the development of IoT-aware CPMs. This is achieved by proposing novel concepts to include the IoT resource perspective in CPMs via configurable IoT resource operators.
- **Chapter 5: Conclusion and Future Works** summarizes the contributions presented in this thesis and concludes the thesis along with providing the future perspectives of our work.

Related Work

Contents

2.1	Introduction	36
2.2	Background	36
2.2.1	Process Modeling Languages	36
2.2.1.1	Business Process Model and Notation	37
2.2.1.2	Configurable BPMN	38
2.2.2	Basic Concepts from IoT Domain	41
2.2.3	Relevant Semantic Models from BPM Domain	43
2.2.3.1	Web Ontology Language for Web Services	44
2.2.3.2	Business Process Modeling Ontology	44
2.2.4	Relevant Semantic Models from IoT Domain	46
2.2.4.1	Internet of Things-Lite Ontology	46
2.2.4.2	Federated Interoperable Semantic IoT Testbeds and Applications Ontology	47
2.3	Support for the Development of IoT-Aware BP Models	48
2.3.1	Allocation of Resources in BPs	48
2.3.2	Allocation of IoT Resources in BPs	50
2.3.3	Synthesis	55
2.4	Semantic Technology in BPM and IoT Domain	56
2.4.1	Application of Semantic Technology in BPM Domain	57
2.4.2	Application of Semantic Technology in IoT Domain	59
2.4.3	Synthesis	60
2.5	Management of Resource Variability in CPMs	63
2.5.1	Support for Developing CPMs	63
2.5.2	Support for Managing Resource Variability in CPMs	66
2.5.3	Synthesis	67
2.6	Conclusion	68

2.1 Introduction

In the previous chapter, we introduced our research problem concerning the integration of the domain of BPM and IoT, especially with respect to the CPMs. In this chapter, we review the literature for the relevant state-of-the-art works corresponding to our research problem. They are: (1) how to uniquely integrate the IoT resource perspective in BP models? (2) how to formally define heterogeneous IoT resources and their allocations along with the semantically enriched BP models to be shared in a common knowledge base? (See Section 1.2.1) and (3) how to integrate the variability induced by IoT resources and their features at the CPM level? (see Section 1.2.2). It helps us to position our work among other existing work in the literature and assists us to justify our problem statement by comparing and contrasting those existing solutions with our contribution.

Before going into the details of the related works, in Section 2.2 we provide some background information needed to better understand our work. Next, in Section 2.3, we investigate existing approaches that support the designing (or modeling) of BPs and the allocation of resource perspective in BP models, especially the IoT resource perspective. In Section 2.4, we focus on the existing approaches related to the ontology-based applications in both BPM and IoT domain. In Section 2.5, we study the existing approaches on developing CPMs and the configuration of process variants. Finally, in Section 2.6 we conclude this chapter. Furthermore, at the end of each section, we provide a synthesis comparing the current research works and their shortcomings with our work, which helps us to clearly position and motivate this thesis work.

2.2 Background

Before investigating the existing works relevant to our research problem, in this section, we present some background information related to both BPM and IoT domain. This information is needed to better understand the work detailed in this chapter and in the following chapters of this thesis. Section 2.2.1 details the modeling languages for developing BPs relevant to our work, while Section 2.2.2 details the basic concepts related to the IoT domain. In Section 2.2.3 and Section 2.2.4, we briefly detail the relevant semantic models from BPM and IoT domain that were reused during the development of our cross-domain ontology detailed in Chapter 3.

2.2.1 Process Modeling Languages

In this section, we briefly present the two modeling languages that have been used throughout this thesis manuscript: (1) Section 2.2.1.1 details the BPMN modeling

language, which is used to model BPs and (2) Section 2.2.1.2 details the C-BPMN modeling language, which is used for modeling CPMs.

2.2.1.1 Business Process Model and Notation

Business Process Model and Notation is a flow-chart based notation supporting the development of BP models. The first version of BPMN (i.e., BPMN 1.0) was released to the public in May 2004. In February 2006, it was adopted as a standard by the Object Management Group (OMG¹). BPMN 1.0 focused on the designing of the process models, while the second version of BPMN (i.e., BPMN 2.0) was enriched to include the executable semantics along with the graphical notation, thus enabling the execution of the processes by BPMS tools. Most prominently, the BPMN 2.0.1² has been formally published by the International Organization for Standardization (ISO³) as their 2013 edition standard: ISO/IEC 19510:2013. The latest version of BPMN is 2.0.2, which was published in January 2014.

Overall, BPMN provides a set of graphical notations that are understandable by different stakeholders in any business setting. These stakeholders are: (1) *Business Analysts*, who develop the initial drafts of the process models, (2) *Technical Developers/IT Architects*, who implement the underlying technology to enable the execution of these processes, and (3) *Process Owners/Business People*, who manage & monitor these processes to provide the desired value to the end customer [11]. Furthermore, due to its ease of use, standardization, availability of tools and features such as extensibility, to name just a few, BPMN has become popular and is widely adopted in both academia and industry [74].

The set of elements provided by BPMN can be grouped into four categories: *Flow objects*, *Connecting objects*, *Swimlanes* and *Artifacts*. *Flow Objects* are a set of elements that assist in modeling the behavior of a BP or in other words, the control-flow perspective of a BP. These elements are Activity, Event, and Gateways, graphically represented in Figure 2.1. An *Activity* represents a unit of work that needs to be completed. Often, it is also called a *task* and is graphically represented as a rectangle with rounded corner. An *Event* consists mainly of three types: *Start*, *Intermediate* and *End* events. The start and end event represent the beginning and end of the process, while the intermediate event represents anything that takes place while a BP is being executed. Graphically, an event is represented by a circle. *Gateways* also represent the control-flow behavior in a process model, especially the decision-making step in terms of the splits (divergence) and joins (convergence). In BPMN, there are three main gateways (also called connectors) for modeling a decision route: *AND* (parallel gateway), *OR* (inclusive gateway),

¹<https://www.omg.org/index.htm>

²<https://www.omg.org/spec/BPMN/2.0.1>

³<https://www.iso.org/standard/62652.html>

and *XOR* (exclusive gateway). There do exist some specialized gateways such as complex or event-based gateways, but they can be mapped to the three main types of gateways mentioned before. The *Flow Objects* are connected to one another via *Connecting objects*, which consist of three main elements. They are: (i) *Sequence flow*, which determines the order of execution of the activities, (ii) *Message flow*, which determines the message flowing between pools, and (iii) *Association*, which enables association of data objects to an activity or a flow element.

The *Swimlanes* consists of *Pools* and *Lanes* elements that help to model a set of activities as a group, where each group has a specific role associated with them for being executed by a specific resource or an actor (resource perspective). The *Artifacts* consist of the *Data object* that allow the modeling of the data perspectives in the process, i.e. the information regarding the data involved in an activity.

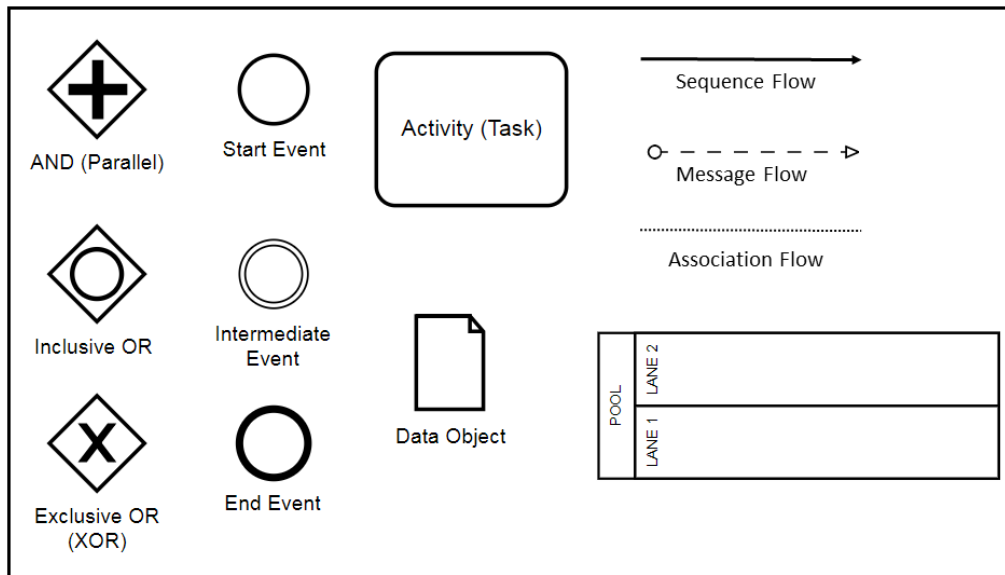


Figure 2.1: Graphical representation of BPMN elements

2.2.1.2 Configurable BPMN

Configurable BPMN (C-BPMN) is an extension of BPMN that includes concepts and elements needed to represent the variability in process models. C-BPMN is widely used to develop CPMs, which are an integrated representation of a family of processes in a given domain [8]. C-BPMN uses variation points (configurable elements) to capture the differences among the process variants (similar to techniques from Software Product Line Engineering) [62]. It maintains a clear distinction between the *commonalities* (i.e., parts shared by all process variants) and *variability* (i.e., parts specific to certain process variants) in a process family.

Figure 2.2 represents the common and variable parts of two different sales process and their aggregated representation in a CPM via a variation point. These modeling techniques allow sharing of knowledge and best practices, which enables the analysts to develop processes based on various guidance and rules (options) provided in these models (at design-time) [8, 62]. In literature, various languages (other than C-BPMN) exist for modeling configurable processes such as configurable Event-driven Process Chains (C-EPCs), UML ADs [62]. In this thesis, we work on CPMs using C-BPMN as it extends BPMN, which is the most popular modeling language in both academia and industry due to standardization, tool support and features such as easy extensibility [74, 75].

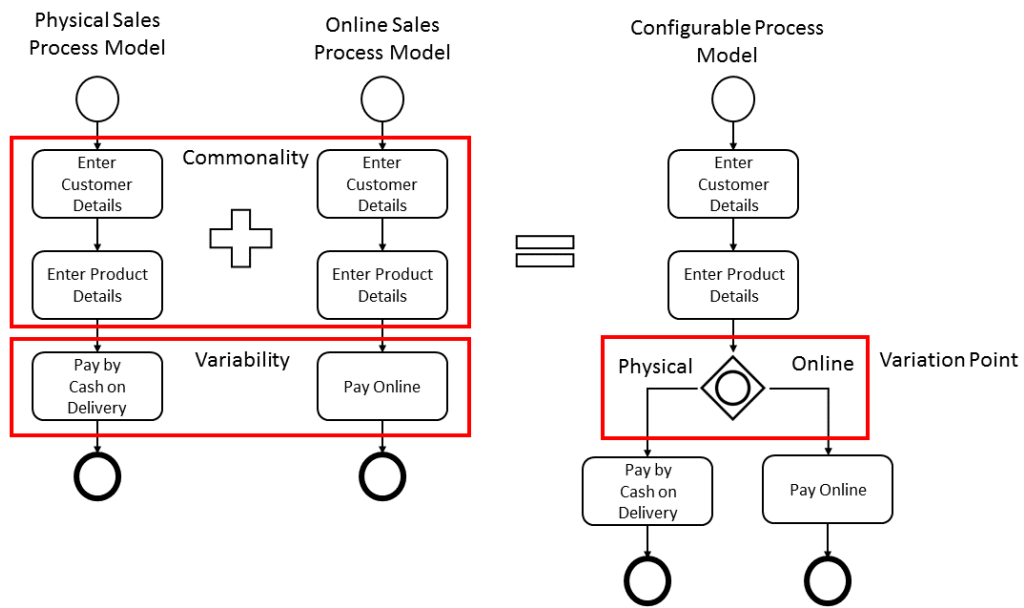


Figure 2.2: Process models having common and variable parts

C-BPMN consists of two configurable elements, i.e., activities and gateways (connectors), which are modeled with a thick line. These elements can be included, i.e., configured to *ON* or excluded, i.e., configured to *OFF*, depending on the specific business requirements. During the design-time, a configurable gateway is configured by a process designer to restrict its behavior, which is based on how it is configured, i.e., (1) changing its type e.g. OR^c to XOR , and/or, (2) restricting its incoming and outgoing branches [8]. Table 2.1 details the constraints on the configuration of the gateways, wherein a configurable gateway is denoted by a $[type]^c$, while a classical gateway is denoted by a $[type]$. Each row in the table corresponds to an initial type of the configurable gateway that can be transformed or configured to one or more types in the columns. For example, the OR^c gateway can be configured to any type, while in an AND^c gateway remains unchanged. The last column marked as Seq. represents a *Sequence flow*. While the gateways

can be transformed from one type to another the behaviors such as a join behavior or split behavior cannot be changed.

	<i>AND</i>	<i>OR</i>	<i>XOR</i>	Seq.
<i>AND^c</i>	✓			
<i>OR^c</i>	✓	✓	✓	✓
<i>XOR^c</i>			✓	✓

Table 2.1: Constraints regarding the configuration possibilities of the gateways

For instance, Figure 2.3 represents a configurable process on the left-hand side and a customized process model in the right-hand side. The configurable process contains a configurable OR (*OR^c*) gateway connecting an activity *A2* and a configurable activity *A3*. A business designer configures the *OR^c* into an *XOR* and keeps the *A3* activity into the process model. Thus, after choosing the configurable elements and the type of transformation need, specific variants can be derived by doing the transformation or removing the excluded nodes and edges based on algorithms such as presented in [8, 75].

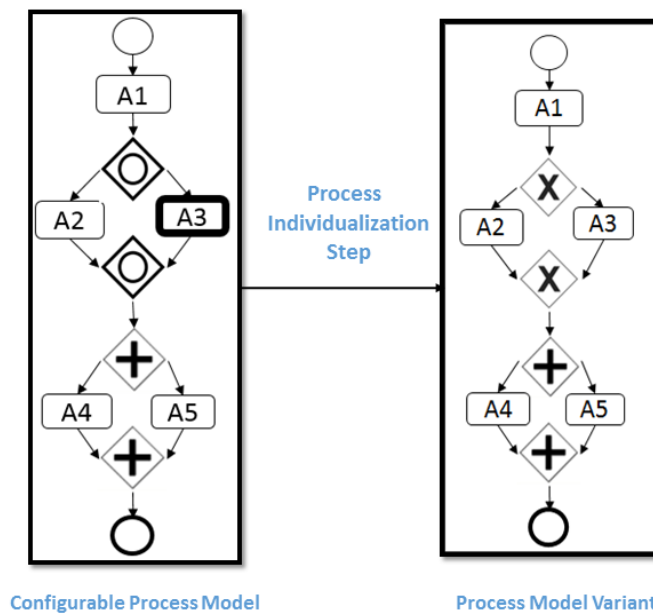


Figure 2.3: Configuration of activity and gateways

2.2.2 Basic Concepts from IoT Domain

The inception of the term IoT dates back to the year 1999, when Kevin Ashton⁴ from Massachusetts Institute of Technology (MIT) put forth the idea of adding RFID tags to everyday objects [76]. Initially, the IoT revolution started with the use of RFID tags, sensors and sensor network for tracking objects especially in the domain of Supply Chain Management (SCM). However, people in both research and industry quickly realized the potential of IoT, which goes beyond tracking objects and thus started to connect IoT devices together for various scenarios. Today, many of these IoT devices enable interactions with the physical world by exposing their functionality via standard services. They are considered to be one of the key technology enablers for fostering the vision of a smart world, which comprises of smart objects, smart logistics, smart manufacturing (Industry 4.0), to name just a few.

IoT comprises of connected devices such as Sensor, Actuators, Tags (e.g. RFID), and smart objects, which supports the creation of a smart (intelligent) environment. Figure 2.4 depicts these IoT devices, which can either be used as simple devices or can be aggregated to create a complex device such as a robotic arm. When the (data-based) intelligence generated from these devices is applied for making successful inferences, it offers a huge potential to change everyday life. Additionally, it allows decision makers to have superior transparency and value-added understanding of their complete product lifecycle. Over the last decade, the application of IoT became more prominent as a result of the technological improvement in the hardware technology, making these devices and the underlying network more available, and affordable for mass usage [77].

In the domain of IoT, the heterogeneity is not only introduced by the large availability and choice of numerous devices but also due to the use of different types of networks needed to connect and communicate with these devices as illustrated in Figure 2.5. In fact, to efficiently consume and manage the deployed IoT resources in context of a BP, there is an evident need to clearly grasp the fundamental concepts in IoT such as topology of network, power usage, bandwidth, intermittent connectivity [31] along with the underlying infrastructure, i.e., Cloud, Fog or Edge computing [78], used for deployment and management of the IoT devices. Some of these concepts are:

- *Power Usage*: Devices consume a considerable amount of power while transmitting data, particularly over long ranges.
- *Bandwidth*: The rate of data transmission depends on the capacity of the

⁴Kevin Ashton: “I could be wrong, but I’m fairly sure the phrase “Internet of Things” started life as the title of a presentation I made at Procter & Gamble (P&G) in 1999”. RFID Journal, June 2009 (URL-<http://www.rfidjournal.com/articles/view?4986>)

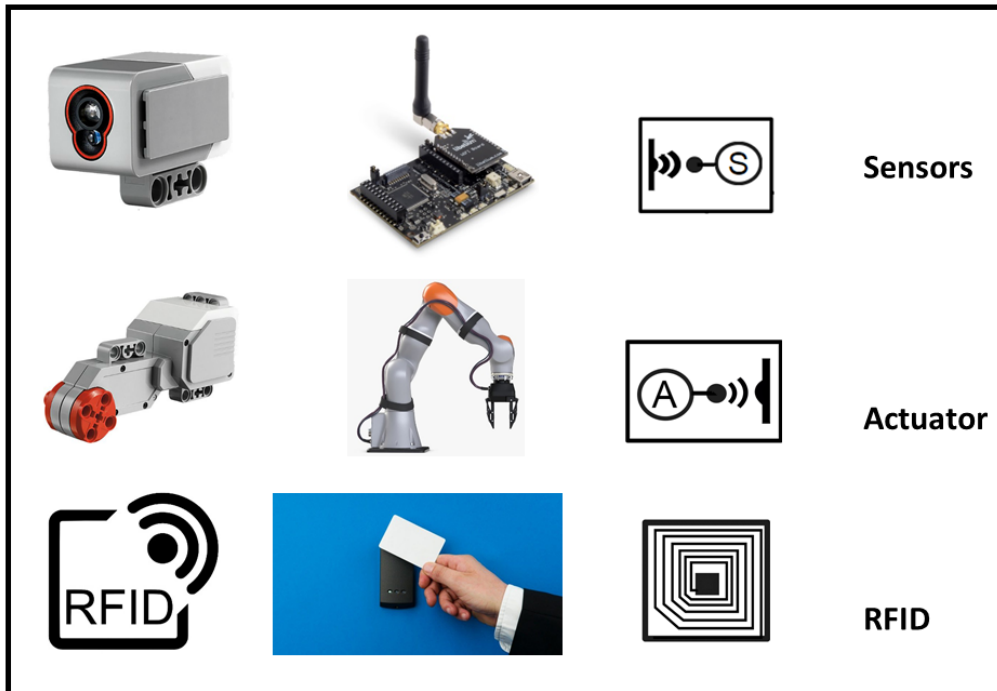


Figure 2.4: Types of devices in the IoT domain

network, and parameters such as the volume of data (raw or aggregated), number of devices, connectivity (constant stream or intermittent bursts of data), the packet size of the networking protocol, to name just a few.

- *Intermittent Connectivity*: To conserve power and bandwidth, devices connect and transmit data periodically (rather than continuously). However, other situations such as an unreliable network or issues with the quality of service (e.g., interference on a wireless network using a shared spectrum), hamper the connectivity.

Furthermore, to illustrate the heterogeneity of the communication network let us take the example of various networks provided by the *Orange Telecommunication*⁵, which is one of the biggest telecommunication providers in France. Figure 2.5 represents the different types of IoT Network solutions⁶ such as LoRaWAN, Cellular 2G/ 4G or Mobile IoT (LTE-M), which a customer may subscribe to based on their specific needs such as data transmitted, power usage, coverage, to name just a few. For instance, the LoRaWAN network has a very low power consumption but the maximum throughput is just 50 kbps. While the power consumption of the Cellular 4G is on the higher side but it also provides

⁵<https://www.orange.fr>

⁶<https://partner.orange.com/orange-iot-networks/>

a throughput in Megabits per seconds (150 Mbps) (see Figure 2.5 obtained from Orange Telecom, France).

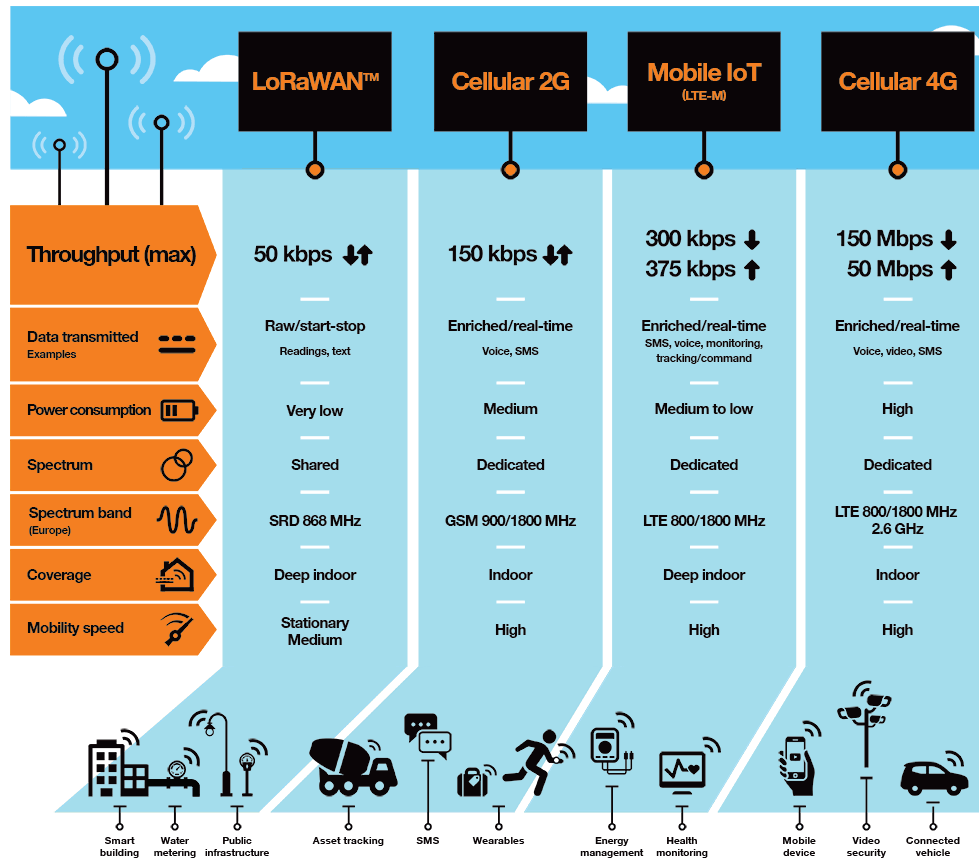


Figure 2.5: Example of various types of network for connecting IoT devices

Overall, the efficient use of IoT resources calls for inclusion of such (technical) information in the process models at design-time. This will ensure proper usage, deployment, and management of IoT resources during the implementation and deployment phase of the BPs.

2.2.3 Relevant Semantic Models from BPM Domain

In this section, we briefly detail the semantic modes from BPM domain relevant for this thesis work. Following the best practices, various concepts from these ontologies were reused during the development of our cross-domain ontology integrating the BPM and IoT domain. Section 2.2.3.1 briefly details the OWL-S, while Section 2.2.3.2 briefly details the BPMO ontology.

2.2.3.1 Web Ontology Language for Web Services

The Web Ontology Language for Web Services (OWL-S⁷) [3] is built on top of the Web Ontology Language (OWL⁸) for providing semantic definition of Web Services. Developed by the Web-Ontology Working Group at the World Wide Web Consortium (W3C⁹), it uses the basic elements of the Resource Description Framework (RDF¹⁰) language to provide a structure for describing identified information in the Web. It allows users and machines to discover, invoke, compose, and monitor the services offering Web resources in an automatic manner. The OWL-S contains the process ontology, which describes the mechanism to interact with a service considering it as a process. Figure 2.6 sourced online from the OWL-S website, illustrates the process ontology modeled in OWL language having its elements, i.e., classes, properties, and axioms. They have specified three main types of processes:

- *atomic processes*, representing the actions a service may perform within a single interaction
- *composite processes*, representing the actions requiring multiple actions and/or multistep protocol
- *simple processes*, representing multiple abstract views of the process.

OWL-S providing an overly narrow view on web services and has conceptual ambiguity, thus it is not suitable to model heterogeneous BPs directly.

2.2.3.2 Business Process Modeling Ontology

Business Process Modeling Ontology (BPMO) [48] enables the representation of various elements from different BP modeling languages such as BPMN [69], EPC [79] and BPEL [33] by providing an abstraction of their modeling notation [80]. It was developed for managing heterogeneous process modeling languages in the context of the European project, Semantics Utilised for Process Management within and between Enterprises (SUPER¹¹) project. Since then it has been used in various other research projects. BPMO includes concepts such as process, task, gateways, which are common for all modeling languages. It has been successfully applied in various works involving representation and comparison of heterogeneous BP models [72]. The core concepts involved in BPMO are:

⁷<https://www.w3.org/Submission/OWL-S/>

⁸<https://www.w3.org/standards/techs/owl>

⁹<https://www.w3.org/>

¹⁰<https://www.w3.org/RDF/>

¹¹<http://projects.kmi.open.ac.uk/super/>

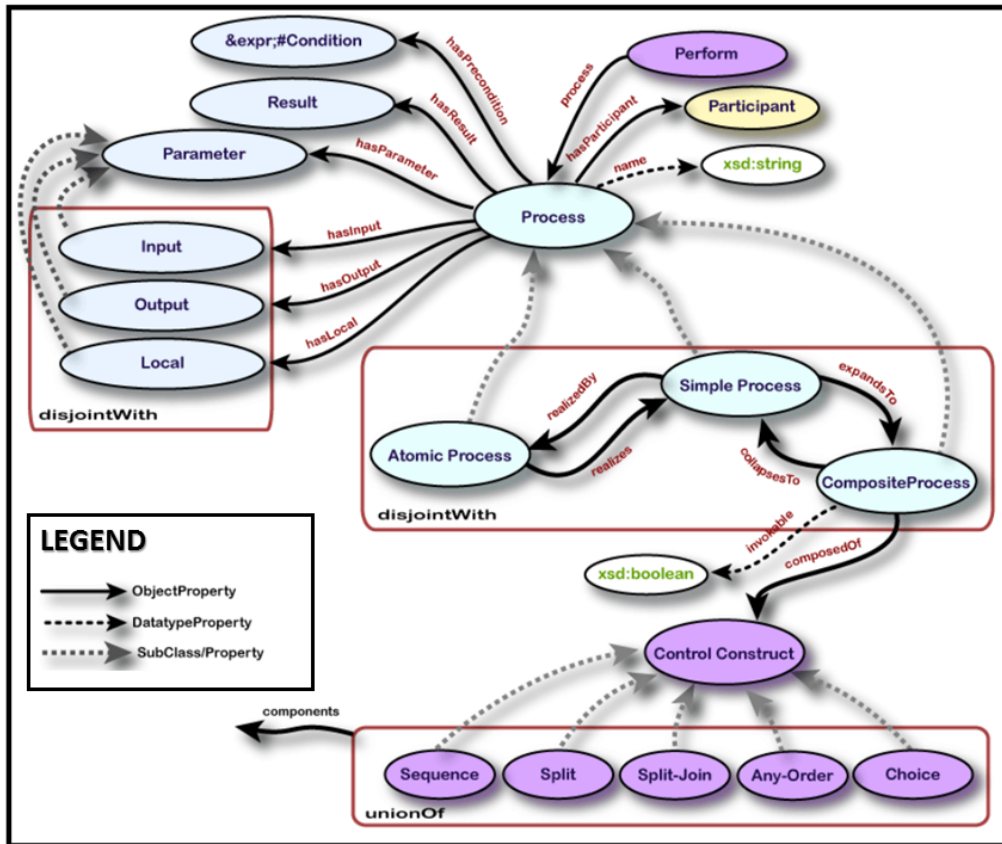


Figure 2.6: The process ontology in OWL-S [3]

- *WorkflowElement*, representing a model element in a BP model
- *Process*, representing an abstraction of a BP model
- *Task*, representing an atomic unit of work or activity in a BP model
- *GraphPattern*, representing the connection between different workflow elements in a BP model

The functionality of a *WorkflowElement* is represented through an object property *hasBusinessDomain* to a concept in a domain-specific ontology that formalizes the business understanding of BP in that given domain. In our work, we use the BPMO ontology (see Chapter 3) to model the basic underlying concepts and relationships from the BPM domain into our semantic model. Furthermore, the work presented in this thesis is focused on the use of BPMN as the primary modeling language. However, the use of BPMO ontology for the development of our semantic model helps in application of this semantic model to other process modeling languages such as EPC.

2.2.4 Relevant Semantic Models from IoT Domain

In this section, we briefly detail the semantic models from IoT domain (similar to Section 2.2.3) relevant to our work. These IoT specific ontologies were reused during the development of our cross-domain ontology detailed in Chapter 3. Section 2.2.4.1 briefly details the IoT-Lite Ontology, while Section 2.2.4.2 briefly details the FIESTA-IoT Ontology.

2.2.4.1 Internet of Things-Lite Ontology

Due to the proliferation of IoT, several research projects such as FIWARE¹², FIESTA-IoT, to name just a few, have been working towards the development of different semantic models for solving the issue related to heterogeneity in the IoT domain and to assist in achieving interoperability between various devices. Following the best practices for ontology engineering, many of the semantic models reuse pre-existing models such as SSN ontology [61] (one of the popular models for sensors). However, due to its complexity and inclusion of non-essential components, SSN model is heavy to query and process, plus difficult to use, especially by non-experts [60]. Some well-known projects that use and extend SSN are, IoT-A [81] and IoT.est¹³ [82]. But, their semantic models are also overly complex for fast user adaptation and responsive environments [60]. Thus, Bermudez-Edo et. al [60] presented *IoT-Lite*¹⁴ ontology, a lightweight semantic model for IoT domain (instantiation of SSN) using concepts from IoT-A reference model. Its lightweight makes it appropriate for real-time device discovery by allowing fast queries on it. Additionally, as it is a *meta* ontology, it can be extended for representing IoT concepts in various domains. The work on IoT-lite was partly supported by EU FP7 FIWARE project and partly by the EU H2020 FIESTA-IoT project. It contains the following core concepts:

- *IoTLite:Service*, describes the functionality exposed by an IoT device that is depicted through concept *IoTLite:Device* (detailed in Chapter 3). This service enables a system (or process in our case) to interact with a physical entity (object of interest) represented by the concept *IoTLite:Entity*.
- The *IoTLite:Device* concept has three sub-types, that are: *IoTLite:TagDevice*, *IoTLite:SensingDevice*, and *IoTLite:ActuatingDevice*. These devices measure or actuate a certain physical quantity, such as, temperature or light intensity that is depicted by the *IoTLite:QuantityKind* concept.

¹²<https://www.fiware.org/>

¹³<http://www.surrey.ac.uk/ics/research/internet-of-things/projects/completed/iotest/>

¹⁴<https://www.w3.org/Submission/2015/SUBM-iot-lite-20151126/>

Overall, IoT-Lite is lightweight, provides a common vocabulary allowing discovery and interoperability of IoT resources in heterogeneous platforms and is extensible. As the first step during the development of our cross-domain semantic model detailed in Chapter 3, we used the concepts from the IoT-Lite ontology for modeling the basic underlying concepts and relationships from the IoT domain [83].

2.2.4.2 Federated Interoperable Semantic IoT Testbeds and Applications Ontology

IoT-Lite provides a standard vocabulary allowing discovery and interoperability of IoT resources in heterogeneous platforms, however being a lightweight semantic model it misses various important concepts from IoT domain such as energy cost. Thus, based on the need for having a unified semantic model that could be used in various domains such as healthcare, transportation, to name just a few, the researchers in EU H2020 FIESTA-IoT¹⁵ project developed a comprehensive ontology for representing knowledge in the IoT domain called as the FIESTA-IoT ontology.

It reuses the concepts from other existing ontologies such as IoT-lite, M3-lite¹⁶, SSN, Time¹⁷ and DOLCE+DnS Ultralite (DUL¹⁸) [59]. The FIESTA-IoT semantic model is envisioned to accomplish interoperability issues by using semantic-based technologies to annotate all the information shared by the different platforms. As various test beds have their own devices and data formats (proprietary format), it makes it difficult to achieve interoperability. FIESTA-IoT ontology surpasses other existing IoT ontologies due to three main reasons that are:

- many ontologies are domain-specific and thus cannot be used in cross-domains projects.
- various ontologies miss some concepts making them insufficient to be used for the data provided by various sensors.
- many ontologies do not follow the best practices, thus are hard to correctly interpret and reuse.

In our work [84] (detailed in Chapter 3), we use the FIESTA-IoT ontology to model the IoT concepts including IoT specific features such as energy cost and their relationships into our cross-domain semantic model.

¹⁵<http://fiesta-iot.eu/>

¹⁶<http://www.sensormeasurement.appspot.com/?p=m3>

¹⁷<https://www.w3.org/TR/owl-time/>

¹⁸<https://lov.linkeddata.es/dataset/lov/vocabs/dul>

2.3 Support for the Development of IoT-Aware BP Models

In literature, several existing works have viewed the domain of BPM from different perspectives, i.e., the control-flow perspective, the data perspective, the artifact perspective, or the resource perspective. Among them, the control-flow perspective has been widely researched as it helps to streamline the temporal aspect of the BPs, which is important for timely execution of the processes that are based on certain Service Level Agreement (SLA). Further, to be cost-effective and gain competitive advantage, it is critical for an organization to properly manage its resources, i.e., both humans and machines/non-human (automation intensive processes). While the resources perspective in BPM has been an area of interest for quite some time, most of the work has been focused on the management of the human resource perspective [36, 67]. In other words, there is a lack of work done on the modeling and management of the non-human resources in the BPM lifecycle, especially the IoT resources that have their own specific features. This is due to the fact that IoT resources are heterogeneous and there is a lack of standardization. Thus, this thesis project is motivated towards handling the issues related to management of IoT resources in the BPM domain.

To understand the existing gaps, we review the current approaches that enable the formalization and modeling of resources in BPs, especially IoT resources. For simplicity, we divide them into the following: (1) existing approaches for allocation of resources in BPs in Section 2.3.1, (2) existing approaches for allocation of IoT resources in BPs in Section 2.3.2.

2.3.1 Allocation of Resources in BPs

In the literature, various works have proposed solutions for proper allocation and management of resources in context of the PAIS [36, 40–44, 67, 85–92]. Some of these work proposed to extend the Web Services Business Process Execution Language (WS-BPEL¹⁹), commonly known as BPEL (Business Process Execution Language), with the ability to manage the human resources. BPEL is a standard from the Organization for the Advancement of Structured Information Standards (OASIS²⁰) that supports execution of actions within BPs by interacting with web services. To enable the support for human resource perspective, the OASIS proposed an extension to the BPEL called as the BPEL4People²¹ [85], which is defined as follows: “a specification to introduce the people activity as a new type of basic activity which enables the specification of human interaction in processes

¹⁹https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsbpel

²⁰<https://www.oasis-open.org/>

²¹<https://docs.oasis-open.org/bpel4people/bpel4people-1.1.html>

in a more direct way". This was an important step towards the development of a (human) resource-aware workflow technology that has been used in various research projects.

Russel et al. [86] proposed the Workflow Resource Patterns (WRPs) to support the representation of a resource along with its utilization in a workflow. They grouped these patterns into different categories such as creation patterns, push patterns, pull patterns, detour patterns, to name just a few. For instance, in the creation pattern, they have a Role-Based Allocation pattern (Pattern R-RBA), which helps to specify constraints (at design time) wherein only a resource belonging to a specific role can execute a specific task. In another work, Russel et al. [93] evaluated the BPEL4People and WS-HumanTask extensions to WS-BPEL 2.0 based on the WRPs. Stropi et al. [40,41] proposed the extension of the BPMN 2.0 meta-model to assist the enrichment of the process models with the human resource perspective, wherein these extensions complied with the WRPs. In [94] Stropi et al. proposed an approach to support the aspects and requirements of the resource perspective needed during the development of a PAIS. Their work supported the following aspects: (i) structuring the resource, (ii) distribution of work and (iii) authorization. They also provided a tool implementation to support their work. Pika et al. [42] proposed a framework that enables knowledge extraction from event logs specifically for human resources. Their approach enables the analysis of human behavior with respect to actual process executions. Suri et al. [67] focused on the human perspective in BPM and proposed a monitoring framework to capture the workload on these human resources in domain-specific BPs. Their approach was focused on better management and scheduling of the human resources and their workload. Cabanillas et al. [36,43,44] proposed several works in the context of the resource perspective in BPM. Their work represents the importance of resource management in BPs and the need for evolution of PAIS into PRAIS (Process and Resource-Aware Information System). In [44], Cabanillas et al. proposed RALph (Resource Assignment Language Graph), which is a graphical notation for human resource assignments in BPs that derives its formal semantics from Resource Assignment Language (RAL) [95]. The RALph notation proposes four type of resource entities: Organizational Unit, Positions, Person, and Roles. Next, it defines the Capability entities, which are persons having a specific capability. It has several Connectors (similar to control flow connectors), which are used to express the resource assignment. In RALph, the authors also consider the resource dependencies and the resource-activity dependency. Havur et al. [92] proposed an approach using Answer Set Programming (ASP) to formally specify optimal work schedule while considering the dependencies between the work items and the resource conflict.

While most of the aforementioned work have tackled the human resource allocation, there do exist some work on managing other resources such as computation

and storage in BPs [88–91]. For instance, in [90] Graiet et al. formalized cloud resources in the context of BPs based on Event-B formalism, Hamila et al. [91] formalized the temporal constraints on cloud resources used in processes. Most of these work focus on scheduling strategies and resource allocation algorithms to benefit from the cloud resources and their underlying power of computation and storage. It is evident that the existing resource management perspective in BPM is primarily focused on the human resource perspective and their modeling (or representation) in BPs along with some focus on formalizing the resources such as storage and computation in BPs. Contrary to our work, the work on BP resource allocation lacks the support for an explicit representation of IoT features in the BPs. IoT resources have specific capabilities as they are made up of heterogeneous resource-constrained devices with specific features such as energy, computation, storage, to name just a few, along with different network choices for communication (see Section 2.2.2). Thus, they have a different expectation for Quality of Service (QoS) than other resources such as humans and cloud, which motivates the need to address this gap.

2.3.2 Allocation of IoT Resources in BPs

Due to the shortcomings in the aforementioned approaches to allocate and manage the IoT resource in BPs (see Section 2.3.1), there has been a tremendous growth on the research for integrating the IoT perspective into the BPM domain [4–7, 29, 37, 38, 66, 96–108]. Many researchers have put forth the importance of research in the direction of the integration of the IoT and the BPM domain, which is mutually beneficial to both the domains. Janiesch et al. [4] described how IoT resource management can benefit from the structured nature and mature tools and technology available in the BPM domain. Additionally, to enable the proper use of IoT resources in BPM, there will be a need to enhance and extend the state-of-the-art of the BPM field. For the same reason, they gave fifteen emerging challenges (C1 to C15) as illustrated in Figure 2.7, which needs to be tackled to enable smooth integration of the IoT and BPM domain.

The early approaches in this direction were focused on modeling sensors and sensor networks into BPs, for instance, Gao et al. [96] extended BPMN with concepts from the Semantic Sensor Network (SSN²²), while Sungur et al. [98] extended BPMN with Wireless Sensor Networks (WNS). In another interesting work, researchers involved in the Internet of Things Architecture (**IoT-A**²³) [5, 6, 37, 66] project (funded by the Seventh Framework Program (FP7) from the EU) developed an Architectural Reference Model (ARM) for IoT domain. The aim of this ARM was to provide a set of key building blocks or concepts involved in the IoT

²²<https://www.w3.org/TR/vocab-ssn/>

²³https://cordis.europa.eu/project/rcn/95713_en.html

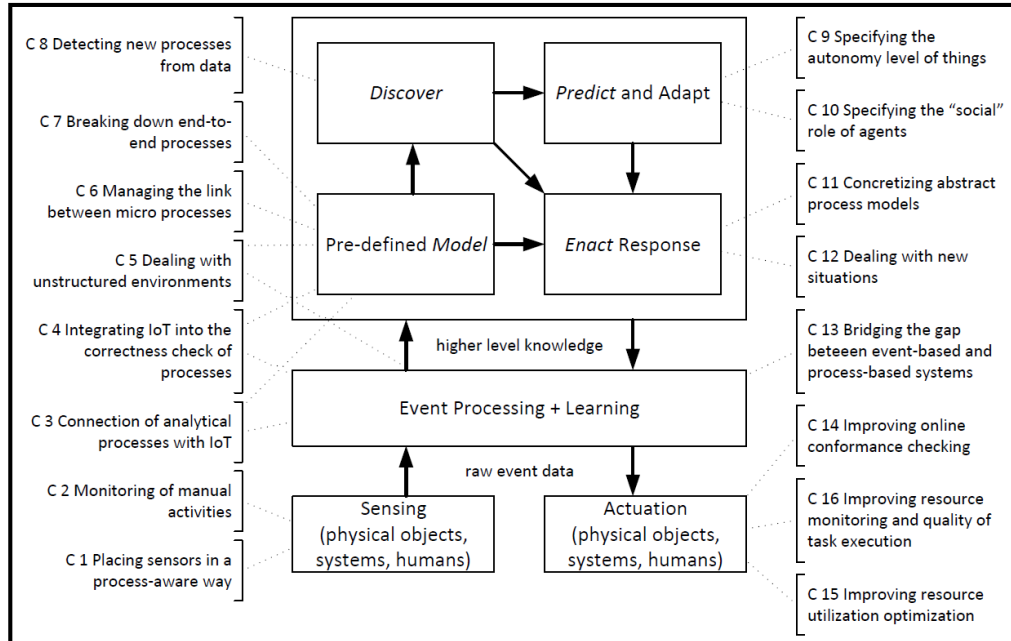


Figure 2.7: High-level overview depicting IoT-BPM interaction [4]

domain. Figure 2.8 illustrates the functional view of the ARM²⁴, depicting the nine Functionality Groups (FG), they are: Management, Service Organization, **IoT Business Process Management**, Virtual Entity, IoT Service, Communication, Security, Application and Device along with the Functional Components (FC) within each of the FG. For instance, in the FG depicting the IoT Business Process Management (relevant to our work), there are two FC, i.e., Business Process Modeling and Business Process Execution.

In the context of the IoT-A project, Meyer et al. [6, 37, 66, 109] introduced an integrated view for the IoT (and also for the Web of Things (WoT)) in enterprise BP modeling (i.e, FG: IoT Business Process Management). Their work paved the path towards a future that includes all layers of networked technology stacks. In [66] they worked on defining the characteristics of real-world aware resources for process modeling. They evaluated various modeling languages such as EPC, UML AD, and BPMN. They found BPMN to be the most suitable for the IoT domain. To model the IoT resources, they extended the Complete Meta-Object Facility (CMOF) for BPMN to introduce new types of task (or activities elements) for IoT Devices. Figure 2.9 depicts the IoT concepts introduced by Meyer et al. such as *IoTDevice*, *IoTAssignment* and their relationships that were included in the BPMN CMOF. These new types of tasks will assist in the modeling of IoT devices (and their services) in the BPs. Further, as illustrated in Figure 2.10 they defined

²⁴http://cocoa.ethz.ch/downloads/2014/01/1524_D1.3_Architectural_Reference_Model_update.pdf

various concepts from the perspective of the IoT devices such as, *Physical Entity*, *IoT Service* and *Native Service*. Overall, their work presented the concepts from the IoT domain and provided an approach for modeling the IoT resources into the BPs. They also implemented a tool for modeling the *IoT-aware Processes*²⁵.

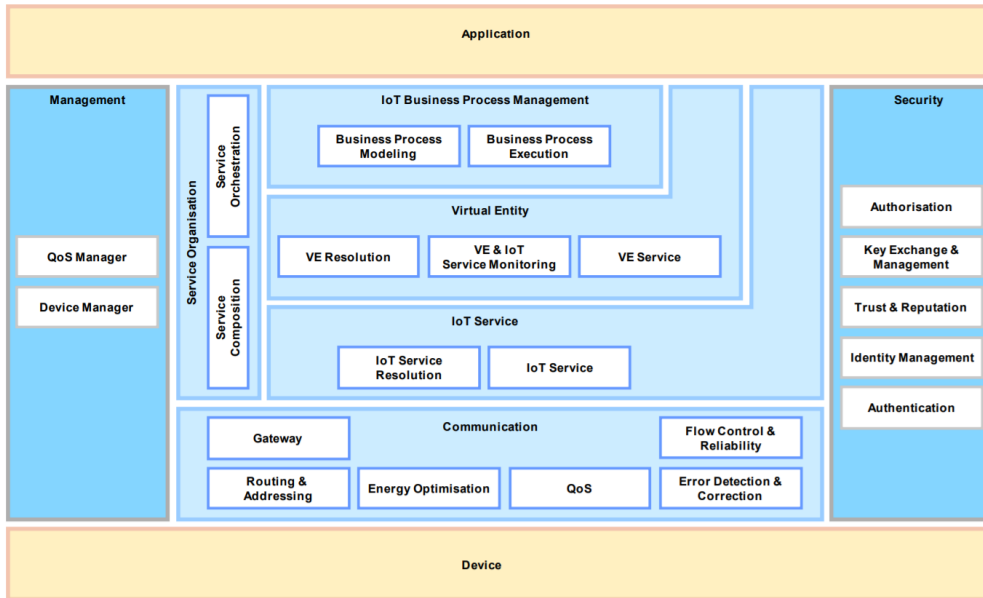


Figure 2.8: Functional view of the IoT Architectural Reference Model [5]

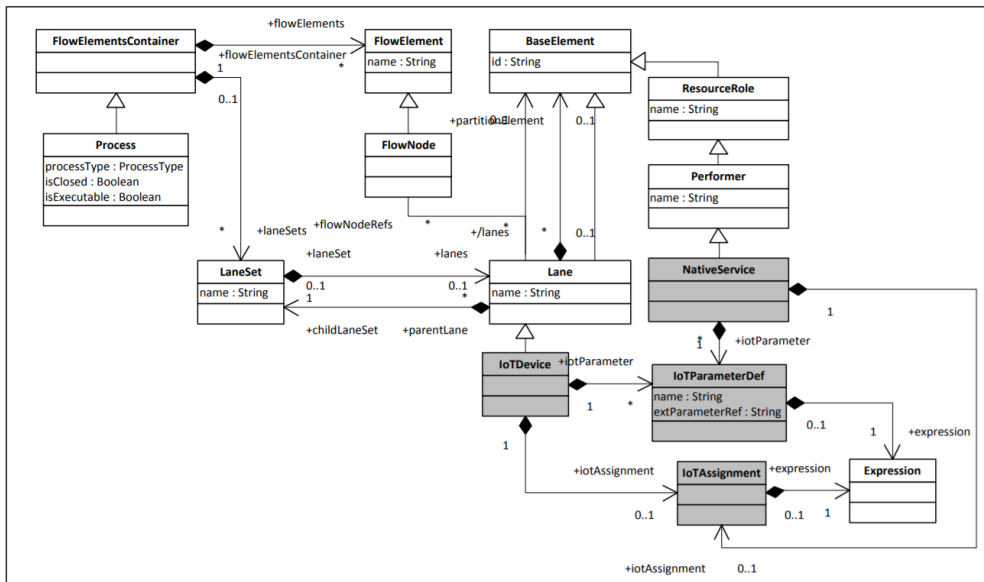


Figure 2.9: BPMN 2.0 extension with IoT concepts [6]

²⁵http://subversion.assembla.com/svn/iot_processmodeler/

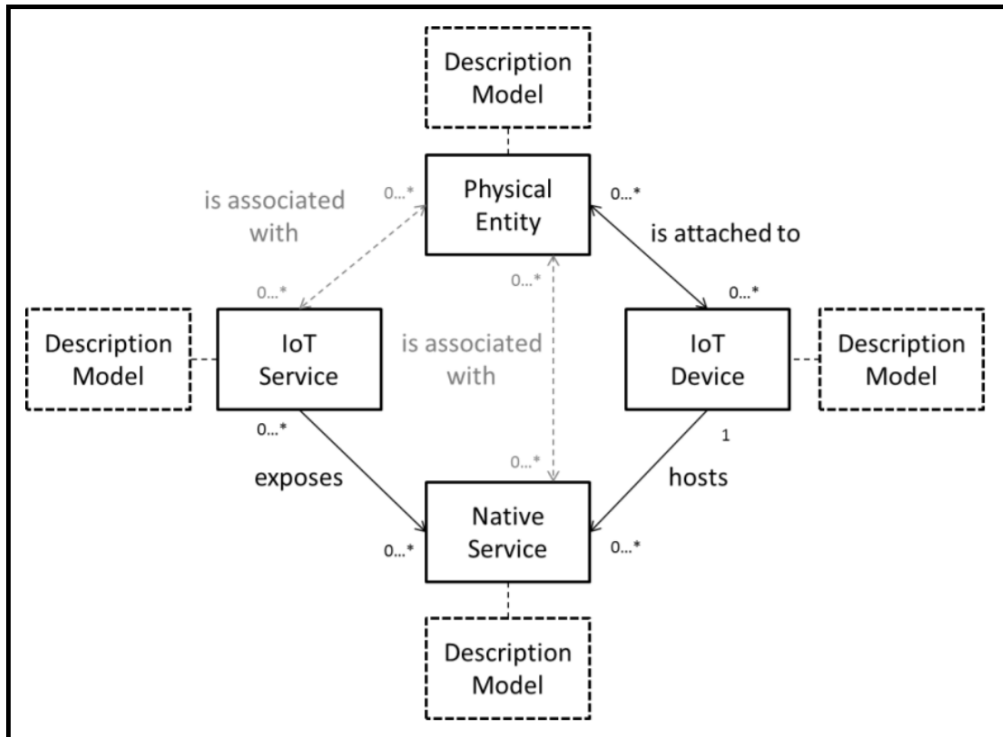


Figure 2.10: IoT concepts relevant for BPM domain [6]

The approach proposed by Meyer et al. [6, 37] suffers from a drawback due to the introduction of new task elements for sensor, actuators, and tags by extending the BPMN meta-model. This is because their approach changes the control-flow model of a BP to accommodate the IoT specific tasks. Their approach promotes the inclusion of the technical know-how of the underlying IoT resources in BPs but it is not relevant for the business stakeholders. Furthermore, it will cause problems during the creation of a CPM (or Reference Model) (see Section 2.2.1.2), which involves merging various process variants having several IoT tasks. These configurable models are needed for knowledge sharing (and reuse) along with the sharing of guidelines and best practices within a multi-national organization. Such organizations have several branches and departments spread geographically that benefit from the use of CPMs [75]. In contrary, our approach involves the association of IoT resources to the tasks, which is achieved by extending the *Resource* element in the BPMN meta-model rather than extending the *task* element (see Section 3.2). This approach helps in preserving the control-flow structure of the process model and is beneficial for the business stakeholders to avoid understanding new task types.

Mass et al. [99] detailed the advantages of managing and using IoT resources via BPMS. Their approach proposed a system design to enable decentralized device-to-device (D2D) based BP execution focusing on addressing the problem

of non-reliable internet connectivity. Tata et al. [101] proposed a formal approach using Petri nets to define the decomposition of process-aware applications in IoT environment. They used Petri nets to provide the correctness of such decomposition and extended the Node-RED tool to create a proof of concept. Wehlitz et al. [100] proposed a concept for developing a service-oriented BPM system architecture to integrate the smart devices as BP resources. Friedow et al. [104] propose an approach to enable coordination of IoT devices using a process engine and using the data generated by the IoT resources in a decoupled manner. They made implementation and wrote adapters to make the IoT devices interact with the BPM engine. Seiger et al. [7] proposed the PROtEUS IoT workflow management system (WfMS) that can handle IoT-related challenges as it includes components for dynamic service selection, complex event processing, human interactions and self-adaptation. Figure 2.11 illustrates the different layers involved in such a WfMS that facilitates the execution of repetitive tasks in IoT and includes features such as flexibility, reuse, configuration and programming of processes in the IoT. Mandal et al. [110] proposed a framework wherein the process engine is subscribed to an event processing platform that manages the event. They extended the BPMN to include specific event types and developed a prototype. Schonig et al. [105] proposed concepts for enabling IoT-aware process modeling and proposed an integrated architecture for involving the IoT and BPM domain. In [106] Schonig et al. proposed a common system architecture to integrate the IoT with the BPM domain to enable IoT-aware BP execution. This framework exploits IoT for BPM by providing IoT data and considering it for interaction in a predefined process model. In [107] Seiger et al. proposed a retrofitting process (and software) based on the MAPE-K feedback loop. This adds autonomous capabilities to existing WfMS's with the option of both invasive and non-invasive retrofitting and allow self-adaptability. Contrary to our work, these approaches for integrating the IoT and BPM domain are mainly focused on the process implementation and execution phase, i.e., how to involve IoT resources in WfMS or BPMS. They do not focus on modeling IoT specific features such as energy costs, quality of data, shareability, to name just a few. These features are needed to be explicitly modeled in the IoT-aware BPs as they impact the SLA of the BPs (during the execution phase). Thus, in our work, we focus on formalizing and modeling these IoT specific features as detailed in Chapter 3.

Maamar et al. [102] proposed the blending of cognitive computing with the IoT resulting in the development of cognitive things (CT) participating in BPs. These CTs will enable reasoning, adaptation, and learning capabilities and will have functional and non-functional restrictions along with price strategies for competition purposes. In [103], Maamar et al. propose to integrate the IoT into BPs by using the principle of storytelling and introducing the concept of Process of Things (PoT). Their approach overcomes the limitation of a dynamic collabora-

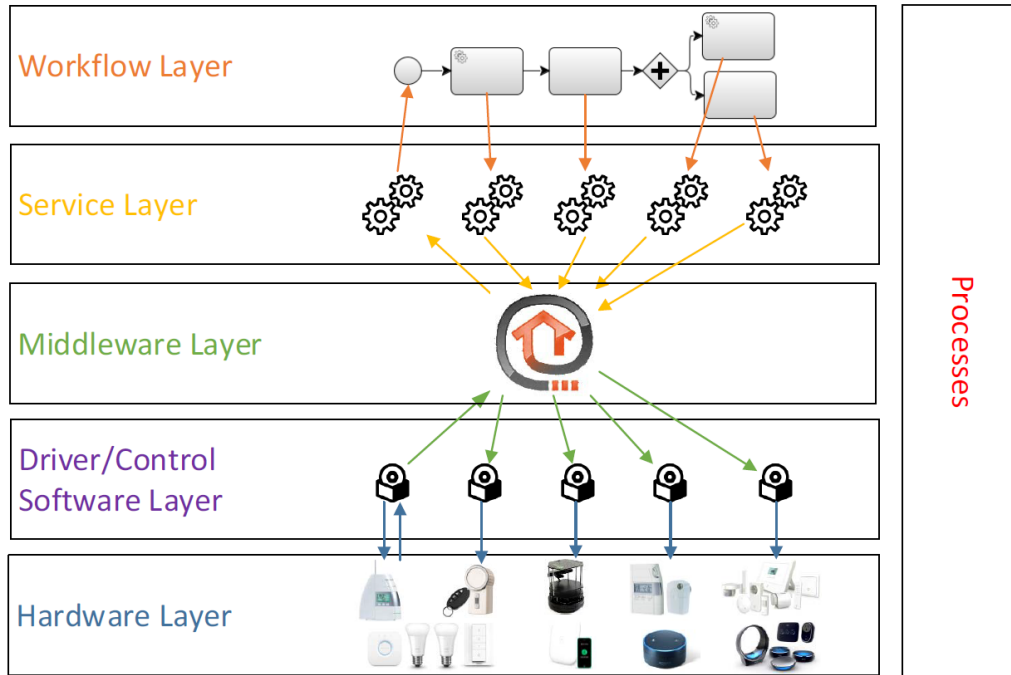


Figure 2.11: IoT architectural layer with WfMS layer [7]

tion that exists in the traditional things-aware BPs. Grefen et al. [29] motivate the idea of an integrated trinity of BPM, IoT and Distributed Analytics. They position their research on inter-organizational collaborative processes that handle physical object such as a container in logistics. They demonstrate the importance of real-time analytics in the management of such processes and the underlying IoT devices. Ye et al. [108] motivate their research using illustration example from the Supply Chain Management domain, wherein they make use of IoT resources in real-time to enable timely decision making and manage the collaborative cross-organizational BPs. They propose the L2L service framework to enable such dynamic collaboration and adaptation of processes. Contrary to our work, these approaches are focused on the use of IoT data (real-time). In this thesis, we do not go into the details of modeling the IoT data and keep our focus on the modeling of IoT specific features such as energy costs, quality of data, shareability, to name just a few in BP models.

2.3.3 Synthesis

Table 2.2 summarizes the existing approaches in the literature based on the criteria relevant to our work. Many of these approaches take into account the resource perspective in BP, wherein some of them consider only the human resource perspective in BPs. Other approaches take into account the IoT resource perspective

in the BPM domain but more from viewpoint of process execution or IoT data perspective. We cluster the above-mentioned approaches based on the criteria they fit the most such as work on human resource perspective, IoT resource or IoT data perspective. In Table 2.2, we use the following nomenclature: “+” to depict that the approaches satisfy the corresponding criteria, “-” to depict that they do not satisfy the corresponding criteria, “+/-” to depict that the approaches partially satisfy the corresponding criteria.

As visible from Table 2.2, the process and the human resources perspective have received an overall good coverage in the literature. These approaches are presented in the first row of the table and were discussed in detail in Section 2.3.1 of this chapter. These approaches [36, 40–44, 67, 85, 86, 88, 90–93] extend the concepts from the BPM domain to include the definition and representation of human resource behavior, which is needed to optimally manage the human resources involved in the processes. There is a need for such approaches as domain-specific processes (e.g. health care) involving humans are scarce and costly. However, our work is mainly focused on modeling IoT resources (and their features) into BPs.

The current approaches in the literature from the context of inclusion of IoT domain in the BPM domain were discussed in detail in Section 2.3.2 of this chapter. They are categorized into three different viewpoints as seen in the Table 2.2: (1) approaches [96, 98] extend the BPs with basic IoT concepts such as Sensor devices and WSN (second row), (2) approaches [5–7, 37, 66, 97, 99–101, 103–107] integrate the IoT resource perspective in information systems, focused mainly on the process execution (third row), and (3) approaches [29, 108] focusing on the inclusion of the IoT specific data (real-time data) in BPs (fourth row). Overall, the work on integrating the IoT resource perspective into process models considers (generic) allocation and orchestration of IoT resources. They do not formalize the IoT resource features (and concepts), and do not consider the heterogeneity in IoT domain. Furthermore, some other work in the literature also involve the use of IoT devices into information systems (IS) [111] but we do not consider them as they are focused on the data perspective and do not view these IS from a process-oriented viewpoint. Contrary to above, our work is focused on formalizing the IoT specific features (along with concepts and their complex relationships) and integrating this IoT resource perspective in the BPs. For instance, the energy-related concept from the IoT domain and its relation to Quality of Information (QoI), which can influence the overall metrics such as SLA of a BP.

2.4 Semantic Technology in BPM and IoT Domain

In this section, we review the literature for the relevant state-of-the-art works corresponding to our research problem, i.e., how to formally define heterogeneous

Approaches	Criteria			
	Process Perspective	Resource Perspective		
		Human Resource	IoT Resource	IoT Data
[36, 40–44, 67, 85, 86, 88, 90–93]	+	+	-	-
[96, 98]	+	-	+/-	-
[5–7, 37, 66, 97, 99–101, 103–107]	+	+/-	+/-	-
[29, 108]	+	+/-	+/-	+
Our Approach	+	+	+	+/-

Table 2.2: Existing approaches with resource perspective in BPM

IoT resources and their allocations along with the semantically enriched BP models to be shared in a common knowledge base? This section helps us to position our work among other existing work in the context of applications of semantic technology in both the BPM and IoT domain. In Section 2.4.1, we review the various research work related to the use of semantic technology in BPM domain. In Section 2.4.2, we review the various research work related to the use of semantic technology in IoT domain.

2.4.1 Application of Semantic Technology in BPM Domain

Over the last decade, various research projects have worked towards the applications of the semantic technologies in BPM domain, thus enabling BPM to evolve into the so-called Semantic Business Process Management (SBPM) [45–47, 49–57, 112, 113]. These works have contributed to the creation of SBPM because of its advantages such as better machine readability/automation, reasoning and the creation of process-oriented knowledge base [45, 46, 49]. In fact, Gassen et al. [57] evaluated the potential benefits of using ontology-based process modeling on the theoretical perspective grounded in cognitive psychology. They performed a quantitative analysis in a controlled experiment that logs every action of the process modeler for analysis. They concluded that providing ontology-based support to the modeler supports the improvement in the design of process models without compromising important aspects such as time consumption and cognitive effort.

Hepp et al. [45] detail the benefits of SBPM and discuss the use of semantic web services for the management of the BPs. In their work, they use Web Service Modeling Ontology (WSMO²⁶) to semantically enrich the BPs. Thomas et al. [47] propose the transformation of process models developed using EPC into semantic process models using the OWL from the W3C. They enriched process models with semantic annotations by mapping the concepts of a formal ontology

²⁶<http://www.wsmo.org/>

to the elements of the process model. This approach has several advantages such as the creation of process-oriented knowledge management, internationalizing of process models (i.e., modeling in various natural languages), to name just a few. In the context of the SUPER project [48,112], Dimitrov et al. [48] used the BP MO ontology (see Section 2.2.3.2) to create semantic annotations for process models modeled in languages such as BPMN and EPC, allowing the modeling of these BPs at an abstraction level. In other words, the BP MO semantic model provides abstractions for heterogeneous process models created in different languages such as EPC, UML AD, BPMN. Pedrinaci et al. [112] discussed the use of WSMO for development of the SBPM. Di Francescomarino et al. [49,114] used semantic techniques to enrich the process models via semantic annotations and assisted in semantic-based process (and data) analysis by performing reasoning (inference) on these semantically annotated processes. Di Francescomarino et al. [114] also proposed to create a logical knowledge base, called Business Processes Knowledge Base (BPKB) by encoding and including the information about the semantically annotated processes using OWL based on Description Logics (DL). This BPKB was developed for querying and reasoning purposes. Riehle et al. [53] proposed an automated, language-independent technique for annotating domain-specific ontology concepts to process model elements at the design-time. This is done by using terminological standardization to create links between the process models and the ontology. They implemented a prototypical artifact to achieve this automated annotation. Such automated integration of ontology concepts and process models saves manual efforts and assist business analysts and modelers from multi-national organizations having offices all over the globe to share a common knowledge and understanding of the BPs. Contrary to our work, all these works are focused on applying semantic technology to BPM domain such as semantic annotations to BP models without considering the integration of IoT domain and its concepts and relationships into the BP models. In other words, they developed the applications of semantic technology in BPs as a silo.

Furthermore, Adamo et al. [56] detail the ontological perspective from the standpoint of the process participants of BPs. In other words, an ontological analysis of BPs modeled in languages such as BPMN, UML AD, or EPC. Lagos et al. [54,55] describe the problem arising because of using generic modeling languages such as BPMN that leads to domain ambiguity and difficulty in management of processes as they evolve (in long-term). They propose that such problems can be mitigated by using domain-specific BPs [67,115,116], which contain domain concepts. However, any changes to the domain concepts by any stakeholders (Modelers/ Business Analysts) needs to be tracked. In [55], Lagos et al. proposed the mapping of domain-specific process models to ontologies using a multi-context systems-based approach, which allows inferring the impacts of changes to the BP models. Their framework supports the long-term governance of BPs in an organi-

zation as it helps to improve the maintainability, reuse, and clarity of these BPs. Hinkelmann et al. [50] presented a hybrid approach to align the IT and Business by annotating the graphical semantic models with concepts from enterprise ontology, thus ensuring consistent modeling. Emanuele et al. [51] proposed a framework to support agility by the means of ontology-aided modeling that would help in fast adaptations for domain-specific modeling languages.

Most of the work related to SBPM, i.e., using semantics for enriching the BPs, is focused mainly on modeling the BP concepts along with approaches to align IT and Business perspective for better management of the IS. In fact, these approaches assist in tackling the heterogeneity in the BPM domain that arises due to situations such as processes modeled in different modeling languages (BPMN, EPC, UML AD etc.), or to promote processes modeled in different natural languages such as French, English, and German (i.e., internationalization). However, contrary to our work, they do not consider an approach to managing IoT resources involved in BPs. These research gaps motivated our work on the development of a cross-domain semantic model for integrating the IoT and the BPM domain and to apply these concepts into the BP models.

2.4.2 Application of Semantic Technology in IoT Domain

In IoT domain, several manufacturers develop their own (heterogeneous) devices having their proprietary data formats, which makes it difficult to achieve interoperability. Nevertheless, the research on IoT domain has gained tremendous momentum as it is considered to be one of the key technology enablers for fostering the vision of a smart ecosystem, such as smart healthcare or smart factories/Industry 4.0 [30,117,118]. Thus, various research projects have proposed the application of semantic technologies in the IoT domain to overcome the problem of device and data heterogeneity [5,58–61]. The IoT-A project [5] provided the key concepts from the IoT domain to be used as the building blocks in various other applications. However, the IoT-A project did not provide a proper semantic model that could facilitate interoperability between different IoT resources. Thus, the FIESTA-IoT [59] project developed a semantic framework for solving the issues related to interoperability in the IoT domain. As mentioned before, the FIESTA-IoT ontology is based on various existing ontologies from the IoT domain such as SSN Ontology, M3-lite ontology, IoT-Lite [60]. Likewise, another semantic model known as the Smart Appliances REFerence (SAREF²⁷) enables interoperability of existing assets, i.e., standards, protocols, or data models in the context of smart appliances domain. It contains concept related to energy profile (power consumption), which is important for optimizing the energy efficiency of

²⁷<https://sites.google.com/site/smartappliancesproject/ontologies/reference-ontology>

a smart building.

Furthermore, to the best of our knowledge, almost all of the semantic models from the IoT domain are focused on resolving the interoperability issues in IoT, without considering the cross-domain relationship between the IoT and BPM domain. Thus, to address this research gap our work is motivated to develop a cross-domain semantic model to bridge the IoT and BPM domain. This comprehensive cross-domain semantic model (see Chapter 3) includes the necessary concepts and relationships to define both the simple and complex concepts from IoT domain such as access cost (AC) and QoI [39] (related to energy and reliability of IoT resources) and its connection to the BPM domain.

2.4.3 Synthesis

As evident from the literature, the use and development of semantic technology in both the IoT and the BPM domain are taking place in silos. In other words, there is no approach that proposes the creation of a semantic model to integrate the complex concepts and relationships from the domain of IoT and BPM.

Table 2.3 summarizes the various existing approaches and follows the same nomenclature of Table 2.2 (see Section 2.3.3). On the one hand, approaches [45–47, 49–57, 112, 113] shown in first row of Table 2.3 are working on the application of semantic technology for BPM as detailed in Section 2.4.1. Most of these approaches use ontologies to semantically annotate the BP models. Thus, allowing the reasoning and querying of these semantically enriched models. Such a semantic enrichment of BP models help to address issues such as heterogeneity in BPs with respect to the modeling languages. This is because the BPs can either be modeled in different process modeling languages or they can have text in different natural languages. While on the other hand, various approaches working on proposing solutions for using IoT resources in information systems also work in silos, without providing any process-oriented viewpoint. These approaches [5, 58–61] detailed in Section 2.4.1 are summarized in the second row of Table 2.3. They are largely focused on the data perspective of IoT to tackle problems such as heterogeneity due to proprietary data formats. These solutions help to infer new and interesting knowledge from the IoT resource data by using various machine learning algorithms.

Contrary to above, in this thesis, we used semantic technology to formalize and integrate the IoT concepts and their complex relationships (i.e, IoT resources) with BP concepts in BP models. For instance, we defined the concepts related to IoT specific features such as energy-cost (aggregation of computation cost, communication cost) and its relationship with concepts such as quality of information, which influences the overall SLA of the BP (during execution phase). The semantic model developed in this thesis work (see Section 3.3) is used to annotate both

Approaches	Criteria		
	Control-Flow Perspective	Resource Perspective	IoT Perspective
[45–47, 49–57, 112, 113]	+	+/-	-
[5, 58–61]	-	-	+
Our Approach	+	+	+

Table 2.3: Existing semantic models for BPM and IoT domain

the IoT and BPM concepts in the process models developed graphically. Further, the semantic model is used to generate an ontology-based knowledge base, which can help to query and reason the underlying IoT devices during process execution.

In Table 2.4, we go a step further to compare and contrast the details of the important semantic models existing in the IoT and the BPM domain, which are relevant to our work. It can be observed that semantic models from the IoT domain are mostly focused on IoT concepts such as sensors, actuators or tags, while a few models also consider the human perspective. Likewise, the important semantic models from BPM domain only consider concepts related to the process (or workflow perspective) along with the inclusion of concepts for human perspective.

Semantic Models	Sensors	Actuators	Tags	Workflow Elements	Energy Cost	Human Resources
IoT-A	+	+	+	-	-	+/-
FIESTA-IoT	+	+	+	-	+/-	+/-
IoT-Lite	+	+	+	-	-	-
IoT-O [119]	+	+	+	-	-	-
M3 Lite	+	+	+	-	-	-
SSN	+	-	-	-	-	-
SOSA	+	+	-	+/-	-	-
DogOnt [120]	+	+	+	-	-	-
SAREF	+	+	+	-	-	-
BPMN	-	-	-	+	-	+
Ontology	-	-	-	+	-	+
BPMO	-	-	-	+	-	+
Our Approach	+	+	+	+	+	+

Table 2.4: Existing semantic models relevant to our work

2.5 Management of Resource Variability in CPMs

Configurable Process Models enable sharing of the process knowledge among various branches of a multi-national organization, which share processes having some commonalities but also some differences [8, 62]. The development of a CPM involves two main steps: (1) finding and collecting all different possibilities (different variations) of a BP that might exist, and (2) merging (or combining) these process models to create a configurable (or customizable) process model containing all the possibilities. In the literature, several approaches exist that support the creation of such CPMs. However, most of them focus mainly on the control-flow perspective without giving much attention to another cause of variability in CPMs such as the variability due to the resources allocated (and participating) in the CPM. In other words, there is a lack of research done towards the management of the resource variability in CPMs and specifically, to the best of our knowledge, there is no existing approach for managing IoT resource variability at the CPM level. This research gap creates a problem to effectively share the knowledge about the processes and the IoT resources involved in them. Thus, rather than receiving some guidance and/or information about constraints about the IoT resources (from a CPM) while customizing (or individualizing) the CPMs into process model variants, these IoT resources and their features shall be modeled without any such support. This may limit expressiveness of the process model and may be time-consuming and error-prone.

In this section, we review the existing approaches in the literature for developing CPMs and for the management of resource variability in CPMs. We divide the available literature into the two following groups: (1) support for developing CPMs (Section 2.5.1) and (2) support for managing resource variability in CPMs (Section 2.5.1). In Section 2.5.3, we evaluate the existing approaches in the context of designing CPMs based on the criteria important to this thesis work.

2.5.1 Support for Developing CPMs

The development and modeling of CPMs is an active area of research in BPM domain. In literature, various existing process modeling languages such as EPC, BPMN, Yet Another Workflow Language (YAWL), UML AD have been extended with special concepts (and elements) to enable the development of CPMs. These modeling languages are: C-EPC, C-BPMN and Configurable YAWL (C-YAWL), to name just a few. These specialized modeling languages (for CPMs) support the explicit representation of the commonalities and the differences by using the variation points (or configurable elements). Thus, various existing research works have developed different techniques for creating (configurable) modeling languages that represent the variation point in the CPMs.

Rosemann et al. [8] extended the EPC modeling language with configuration concepts to develop the C-EPC. In general, EPC has three main control-flow elements: function, events, and gateways. While in C-EPC they introduced two new elements: configuration guideline and requirements, which assist users while choosing the correct choices during the configuration of the process model. Both the guidelines and requirements are a type of constraints that are expressed as logical predicates in form of if-then rules, wherein a guideline is considered as a soft constraint, while a requirement is a hard constraint. Configuration (or customization) of the CPMs is done by restricting the behavior of elements in the C-EPC so as to derive the needed EPC process variant by assigning a configuration choice to a configuration node at the design-time. Figure 2.12 represents a C-EPC process model having two activities modeled in thick lines and showing the guidelines and requirements (i.e., Guideline 1 and Requirement 1) on the configurable element (i.e., configurable activity).

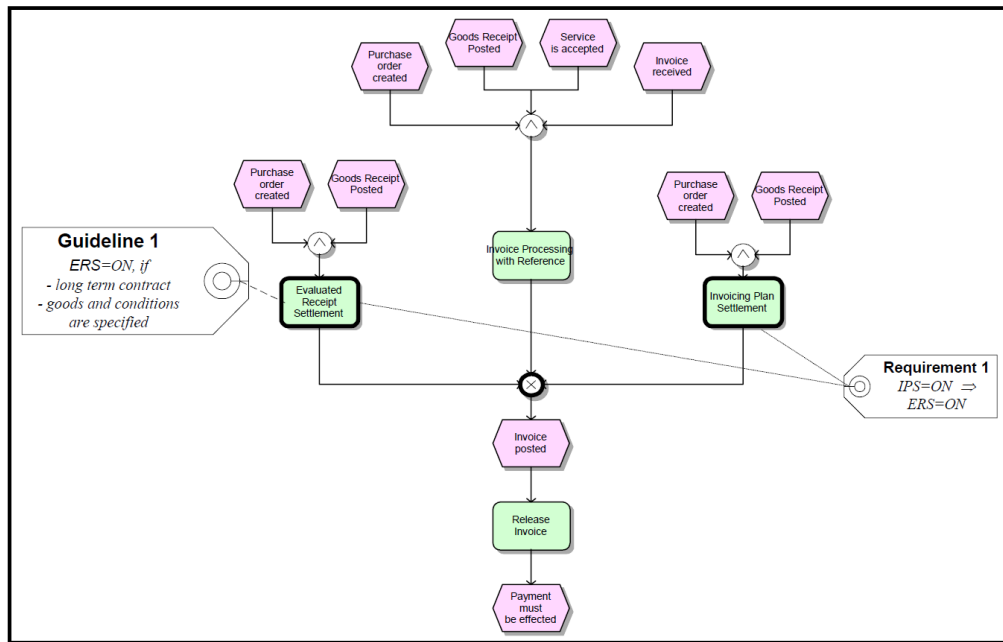


Figure 2.12: An example representing a C-EPC process model [8]

Gottschalk et al. [9] extended YAWL into C-YAWL with so-called *Ports* as the variation points. In their approach, they used the technique of hiding and blocking operators on edges (i.e., transitions) of *Labeled Transition Systems* (LTSs) to restrict and configure a workflow model [63, 121]. In their approach, each task consists of an input port that represents a join of the arcs to enable a task and an output port representing the split of arcs that can be enabled after the task is complete. Figure 2.13 represents a transformation wherein an OR-split connector is configured by activating the one input port (coming from arc “a”), while only

the two output port having the condition “b” and “c” are activated, thus changing the behavior of the split to AND-split.

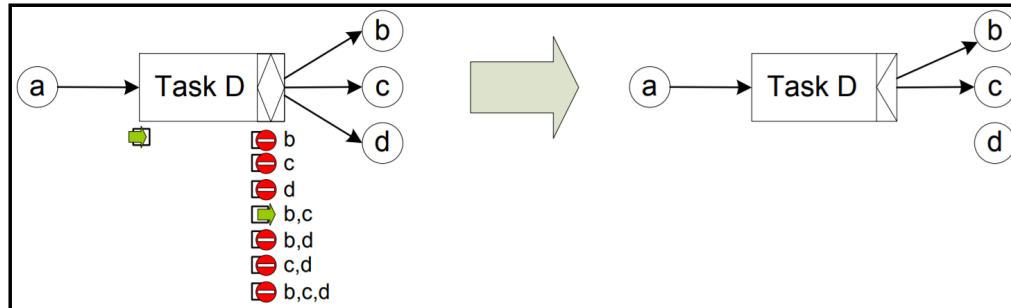


Figure 2.13: Configuration from OR to AND by hiding and blocking [9]

Hallerbach et al. [10] proposed the *Provop* (PROcess Variant by OPTions) approach for modeling and configuration of process variants based on the application of a set of defined operation such as INSERT or DELETE or MOVE fragment, or MODIFY to a reference process model (or a base process model). In general, these change operations occur together and thus they are grouped into the so-called *Options*, wherein each of these options have their own constraints. Figure 2.14 depicts the entire lifecycle of the Provop approach - the first layer consists of the process modeling step having a base model. The second layer consists of various options that are applied to this base model to create the process variants, i.e., the configuration of variants. Once these process variants are developed, they can then be executed via a PAIS such as a Workflow management systems.

Kumar et al. [64] introduced an approach to configure a process template (or a base model) by using configuration rules. The process template is composed of a block-structured tree-like process model. The configuration rule can be used to both restrict or extend the behavior of the process template. These restrictions or extensions are performed via change operators such as INSERT or DELETE. In their approach, the authors formalized the configuration rules and provided an algorithm to individualize the process template using the configuration rules. Their approach is applicable to workflow tasks and does not consider the connectors. However, it facilitates configuration of process variant using change operators, which are proven not to cause any syntactical or behavioral issues to the process structure. Schnieders et al. [122] provided an approach to extend the UML AD with stereotypes marked with `<<VarPoint>>` to depict the variation points in context of the PESOA (*Process Family Engineering in Service-Oriented Applications*) project. These stereotypes assist in identifying the variation points and in the development of variant-rich process models. They can be applied to both UML AD and BPMN. However, in BPMN the annotations can only be applied to activities and their data object, without considering the operators.

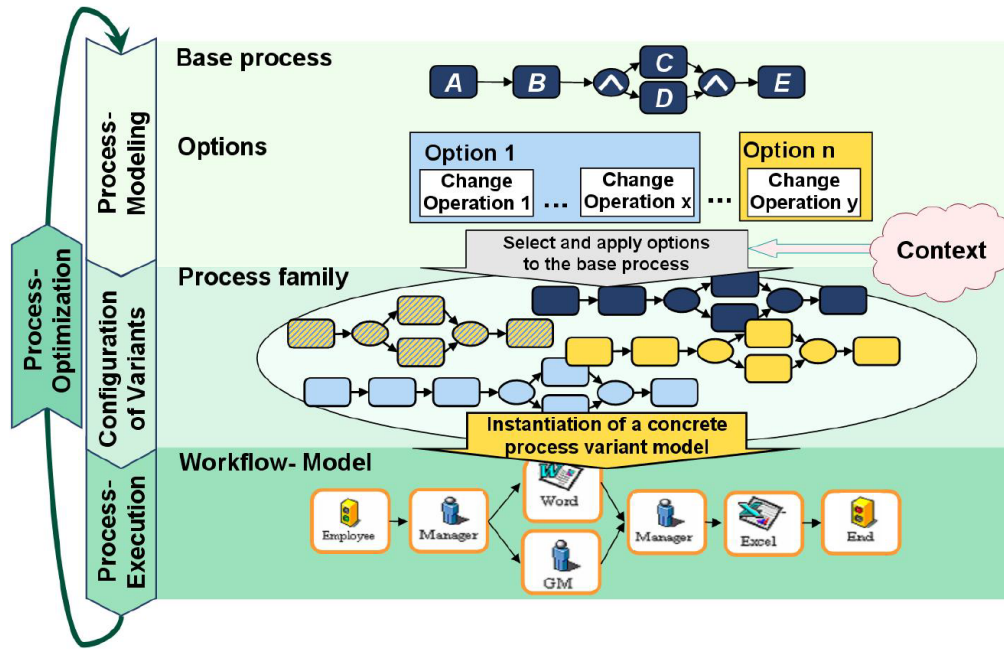


Figure 2.14: Provop process variant lifecycle [10]

In this thesis, we design and model the CPMs using the configurable nodes approach proposed by [8] as it is based on a solid foundation of reference modeling. In our approach, we create a CPM that contains or holds all the possible behaviors (or variations) of the process in question. Thus, the process designer just needs to restrict the behavior of the CPM, rather than including information or content to it. Furthermore, our contribution in Chapter 4, we use C-BPMN (see Section 2.2.1.2), which is an extension of BPMN. BPMN is one of the most important choices for modeling processes and is widely used in both academia and industry due to standardization, tool support and features such as easy extensibility [74, 75].

2.5.2 Support for Managing Resource Variability in CPMs

In literature, most of the research for CPMs is focused on the control-flow perspective. Though, some works such as [62, 64, 65] extended the configuration concepts to include the resource perspective [62], these works are not sufficient enough to handle the features and properties of the IoT domain.

La Rosa et al. [62, 65, 123] proposed the configurable integrated EPC (C-iEPC), which included features for capturing resource, data and physical objects via configurable connectors (based on control-flow perspective) to model the variable allocation of resources. The configuration of these elements is done via configurable connectors (i.e., the control flow perspective). Thus, the authors propose

the modeling of resource variability by associating a process function to a variable number of resources and data objects via these configurable connectors. These connectors have a range parameter that specifies the maximum and the minimum number of elements that can be selected via a configuration choice. Contrary to our work, their work is focused mainly on modeling the human resource perspective and generic non-human resources without any support for handling the variability induced due to the IoT specific features.

As detailed before in Section 2.5.1, Kumar et al. [64] proposed an approach based on templates and rules for creating CPMs. This approach supports the integration of both resources and data perspective. Apart from the control-flow related rules, they also gave the resource related rules, data related rules, and hybrid rules. Their approach included the use of additional information that can detail the type of resource involved e.g, equipment and the data value of parameters associated with such a task. However, their approach does not cover flexible resources selection and is also not suitable for handling the variability induced by the IoT resources and their specific features.

Overall, the literature on developing CPM comprise of only a few approaches that focus on the resources perspective. Most of these approaches are focused on the human resource perspective or handle the resource perspective in CPMs in a very generic way. To the best of our knowledge, there is no proposal that considers the IoT domain and its features such as resource sharing and replication (see Chapter 4), which leads to the creation of multiple IoT-specific process variants. Thus, to assist the proper reuse of process knowledge and to exploit the full potential of CPMs, IoT resource perspective must be integrated into CPMs. To address this research gap, our contribution detailed in Chapter 4 tackles the IoT resource variability by proposing a set of operators, so-called as *Configurable IoT Resource Operators* for integrating IoT features in CPMs. It also provides guidelines and restrictions to assist the customization of these IoT-aware CPMs.

2.5.3 Synthesis

Table 2.5 summarizes the approaches presented in this section for modeling CPMs based on the criteria important to this thesis. They are: (1) Control-flow perspective, (2) Generic Resource perspective, and (3) IoT Resource perspective.

As detailed in Section 2.5.1, most of the approaches [8–10,122] for developing CPMs are focused on the control-flow perspective as visible from the first row of Table 2.5. The resource allocation (and variability) in CPMs is partially handled by some approaches as visible in the second row of Table 2.5. Moreover, in the current state-of-the-art, there is a lack of work for tackling the IoT resource variability at the CPM level. Thus, to address this research gap it is necessary to create concepts for supporting the modeling of IoT-aware CPMs and to enable

management of variability due to the IoT specific features such as shareability as detailed in Chapter 4.

Approaches	Criteria		
	Control-Flow Perspective	Resource Perspective	IoT Perspective
[8–10, 122]	+	-	-
[64, 65]	+	+/-	-
Our Approach	+	+	+

Table 2.5: Evaluation of existing approaches for developing CPM

2.6 Conclusion

In this chapter, we presented the current state-of-the-art with respect to our research problems detailed in Chapter 1. We reviewed various approaches in the literature for developing IoT-aware BPs, i.e., modeling and integrating the IoT resource perspective in BP models along with the approaches to model the CPMs. As the first step, we briefly presented certain background information need to better understand the work in this chapter and in the following chapters. Here, we presented the modeling languages, i.e., BPMN and C-BPMN, which have been used extensively in this work. Next, we introduced the concepts related to the IoT domain, which are needed to understand the IoT domain, which motivates the need to consider IoT features in our work.

In the next sections, we divide the related work based on our research questions into three main groups: (1) formalizing the IoT resource perspective in BPs, (2) application of semantic technology in BPM and IoT domain, and (3) management of IoT resource variability in the CPMs. In the first group, we reviewed the approaches available in the literature from the following perspective: (1) allocation of generic resources in BPs, (2) allocation of IoT resources in BPs. In the next group, we reviewed the approaches from the following perspective: (1) application of semantic technology in the BPM domain, and (2) application of semantic technology in the IoT domain. In the last group, we studied and reviewed the approaches to develop CPMs and to tackle the resource perspective in CPM. We evaluate and compare the existing approaches to understand the missing parts of the approaches in the state-of-the-art, which are (1) lack of modeling support for IoT resource perspective in BPs, especially based on the IoT specific features and issues such as heterogeneity, (2) lack of semantic technology to bridge the IoT and BPM domain, and (3) lack of the management of IoT resource perspective in the CPMs, which is caused by the IoT specific features.

To bridge the above-mentioned research gap, we detail our contribution in the

following chapters. In Chapter 3, we detail our contribution towards development of a cross-domain semantic model that enables the formalization and enriched modeling of the IoT resource perspective in BPs. This semantic model is also used to develop an ontology-based knowledge base, which could assist in reasoning purposes. In Chapter 4, we tackle the issues related to the modeling and management of the IoT resources in the CPMs, as the current state-of-the-art does not support it.

Support for Modeling Semantically Enriched IoT-Aware Business Processes

Contents

3.1	Introduction	72
3.2	Extending the BPMN 2.0 with IoT Resource Perspective	73
3.2.1	Integrating the IoT Perspective in BPMN 2.0	74
3.2.1.1	IoT Specific Energy Perspective in BPMN 2.0	75
3.3	Semantic Formalization of IoT Perspective in BPs	77
3.3.1	Approach Overview	78
3.3.2	Ontology Concepts	79
3.3.2.1	IoT Services & Non-IoT Services	81
3.3.3	Ontology Attributes	84
3.3.4	IoT Resource Properties and Relationships in BPs	85
3.3.4.1	IoT Resource Properties in BPs	85
3.3.4.2	Relationships Between IoT and Other Resources in BPs	89
3.3.5	Managing IoT Resource Conflicts using Strategies	92
3.3.5.1	Constraints on IoT Resources and Task in BPs	92
3.3.5.2	Strategies for Resolving IoT Resource Conflicts	93
3.4	Evaluation and Validation	96
3.4.1	Ontology Validation	96
3.4.2	Support for Developing IoT-Aware BP Models	99
3.4.2.1	Integrating IoT Resource Definitions in BP Models	99
3.4.2.2	Semantic Annotations in BP Models	99
3.4.3	Ontology-based Knowledge Base	100
3.4.3.1	Querying the Knowledge Base	101
3.4.4	Threats to Validity	102
3.5	Conclusion	103

3.1 Introduction

In the previous chapter, we presented the related works to better understand the current state-of-the-art with respect to our research problems detailed in Chapter 1. In this chapter, we present our approach to support the modeling of the IoT resource perspective in the BPs along with semantically enriching these IoT-aware process models. This contribution is motivated to answer the following questions related to the research problem pointed out in Chapter 1: (1) RQ1: How to uniquely integrate the IoT resource perspective in BP models?, (2) RQ2: How to formally define and include the specific IoT concepts along with their relationships into BPs? and (3) RQ3: How to formally define heterogeneous IoT resources and their allocations along with the semantically enriched BP models to be shared in a common knowledge base?

On the one hand, there has been tremendous growth in the application of IoT to offer (data-based) intelligence in various systems. This growth is a result of improvement in the hardware technology that has made these devices better and more affordable for mass production and usage [124]. These devices interact with the physical world and give information about it via a standard service or an application programming interface (API) [6]. They are considered as one of the key technology enablers for fostering the vision of a Smart Ecosystem (such as Industry 4.0 [30]). Nonetheless, these IoT resources are heterogeneous as each of them has its own specific features such as a specific device type, sensitivity level, energy capacity, computation power and/or manufacturers [31]. This heterogeneity leads to the problem of interoperability and increased complexity, thus creating a bottleneck for their application in various domains. On the other hand, these heterogeneous IoT devices must be orchestrated in a specific manner to generate a defined value for an end user (or system). Thus, it is natural to see the IoT devices participating in several collaborative cross-organizational BPs, which orchestrates them along with other resources, i.e., human resources and other enterprise services, to achieve a specific business goal.

For instance, our motivating example in Chapter 1 (see Figure 1.3) depicts one such IoT-aware BP for monitoring a perishable item such as a Chinese Orchid flowers. This process measures a “physical quantity” such as temperature via a specific activity that is associated with a temperature sensor. This sensor interacts with the physical world and provides the information via a software (native service) on the sensor device to the activity executing in the process [66]. Nevertheless, a single process may involve multiple sensors measuring different physical aspects such as temperature, light, humidity, to name just a few. Each of these devices may have specific features and characteristics such as sensitivity of measurement, energy capacity etc. Thus, these heterogeneous devices with their numerous features increase the overall complexity involved in optimal management of the IoT

resource in BPs. Hence, in order to leverage these IoT devices participating in BPs, a PAIS must evolve into a Process- and IoT-Aware Information Systems, by including IoT resource perspective along with its related concepts and relationships in the process models during the design-time. However, even with a growing interest towards the integration of IoT concepts in the BPM domain, there is lack of work for tackling the complexity arising due to this integration. This complexity is caused due to the IoT specific features (properties and behaviors) w.r.t. BPs.

In this chapter, our contribution addresses this research gap by proposing a semantic formalization of IoT resource perspective (including concepts and relationships) in BPs. To achieve this, our approach does the following: (1) provides an extension to the BPMN 2.0 meta-model for integrating the IoT resource perspective, (2) provides a cross-domain semantic model that integrates the IoT concepts and their complex relationships with the concepts from the BPM domain, (3) provides strategies for resolving resource-based conflicts, and (4) provides an ontology-based knowledge base (KB) that includes the information about the processes and the heterogeneous IoT resources. Being machine-readable, the KB assists in reasoning and discovering new relationships.

Overall, the chapter is organized as follows: in Section 3.2, we detail the extension to the BPMN 2.0 meta-model for including the IoT resource perspective. This extension supports modeling and clear representation of the IoT resources (i.e., sensor, actuator etc.) in the BPs. In Section 3.3, we detail our semantic model by describing the ontology concepts, its attributes along with the IoT specific properties and relationships. It also defines various semantic rules (and resource constraints) based on the Semantic Web Rule Language (SWRL¹) formalization to assist in solving resource conflicts using certain strategies. This semantic model is used for developing semantically-enriched IoT resource description in BPs. In Section 3.4, we detail the evaluation and validation of our approach along with the implemented proof of concept to support the development of IoT-aware BPs and its semantic (annotation) enrichment. Finally, we conclude this chapter in Section 3.5.

The contributions presented in this chapter were published in a conference proceeding [83] and in a journal article [84].

3.2 Extending the BPMN 2.0 with IoT Resource Perspective

Recently, several research initiatives have proposed approaches to develop the concepts for IoT domain such as the IoT-A project [5, 125], FIESTA-IoT project [59],

¹<https://www.w3.org/Submission/SWRL/>

FIWARE project, to name just a few. While most of these works have focused solely on the IoT domain to solve problems such as interoperability, there has been a lack of (cross-domain) work for integrating the IoT and BPM domain. In the context of the IoT-A project, Meyer et al. [37, 66, 77, 124, 126, 127] worked on proposing an approach for modeling IoT-Aware BPs, i.e., integrating the IoT perspective in BPs. However, their approach extended the BPMN 2.0 by introducing new types of IoT-specific tasks such as Sensor Task, Actuator Task. This approach has a specific problem that it changes the control-flow perspective of the existing process models by including the IoT perspective via new task types. In other words, with the use of new task types, their approach updates the BP models with the technical details about the underlying IoT devices, which is important during the (technical) process implementation phase, but not relevant to the business stakeholders. Furthermore, this approach hinders the ability to reuse the BPs through the development of the CPMs, which is done by merging the different available variants of a BP. These CPMs are an instrument to share process knowledge and are employed by various multi-national organizations to enable flexibility in the dynamic business environment. In contrary, our approach is based on including the IoT resource perspective in process models by extending the *Resource* element of the BPMN 2.0 meta-model. This approach avoids the problem of making a change to the control-flow perspective of a BP model while providing the necessary technical details about the underlying IoT resources needed during the implementation of the BPs. In Section 3.2.1, we detail this approach for including the IoT perspective in BPs.

3.2.1 Integrating the IoT Perspective in BPMN 2.0

Business Process Model and Notation (BPMN) is one of the most commonly used modeling notation in both industry and academia (see Section 2.2.1.1). BPMN 2.0 is the latest version and includes a very simplistic description of a *Resource* concept required for interacting with a resource in a BP, as detailed by the Object Management Group (OMG) [69]. This resource element is quite abstract and is mainly designed to depict the human resources participating in a BP. In other words, BPMN 2.0 does not provide any formal definition of resources' belonging to the IoT domain. Thus, taking the advantage of the extensibility of the BPMN meta-model, we extend the *Resource* element in BPMN 2.0 with an element called *ResourceExtension* as detailed in the class diagram in Figure 3.1. From the BP point of view, a *ResourceExtension* can include two types of resources: (1) a *HumanResource*, and (2) a *Non – HumanResources* (IoT Device, network). The latter represents the elements from the IoT domain that are mainly Sensors, Actuators or Tag devices, along with the underlying Network. Additionally, based on the business process requirements (and the availability of the resources), both resource types, i.e., the *Human Resource* and the *IoT Resource* can need the

assistance or services of one another.

This approach for creating a resource extension to enrich BP models with detailed resource properties is inspired from the work of Stroppi et al. [41]. The elements *BPMN:Resource* and *BPMN:ResourceParameter* already exist in the default BPMN 2.0 meta-model. In our work, we extend the *BPMN : Resource* via a *ResourceExtension* element, which is composed of one or more *ResourcePrivileges* based on whether resources are to be kept private during a process execution or shared. *ResourcePrivileges* also depict if the resource is in a particular state, i.e., allocated or not allocated (see Section 3.3.3). In Figure 3.1, the *Action* element represents a work item that is carried on a resource such as to execute or actuate an IoT resource, to start or terminate a resource, to name just a few. These elements adapted from [41] are shown in blue color. Furthermore, Figure 3.1 depicts the elements from the IoT domain (*Non – HumanResource*), i.e. the *NetworkResource* and the *IoTDevices*, represented via orange color. The *IoTDevices* concept is further sub-divided into *TagDevice*, *Actuator* and *Sensor*, wherein each of these concepts (or BPMN elements) depict a device from the IoT domain that is used to achieve a specific functionality in the real world. In this work, we represent the network and the IoT device class as a subclass of non-human resources to depict that these two classes belong to the non-human resources, while there may be other non-human resources in context of BPM domain. For instance, the traditional SOA based services, Cloud computing or Edge resources, are some of the examples of a non-human resource that may be involved in a BP. However, in this thesis, the focus of our work is on the specific features related to the IoT devices and the types of networks that may be involved in an IoT-aware BP.

The class diagram in Figure 3.1 is translated into an XML Schema Definition (XSD) format, wherein all these concepts are included in the *Semantic.xsd* document. This document when imported in BPMN 2.0, allows the representation of the IoT resources in the process models.

3.2.1.1 IoT Specific Energy Perspective in BPMN 2.0

IoT devices such as sensor and actuators use energy (electrical power) for processing the information about their surroundings and sending them over a communication channel (a network). However, these IoT resources have a limitation in terms of both the computation power and energy. Thus, the services provided by IoT devices do not have the same expectation such as QoS as that from the classical enterprise or Cloud-based services. Hence, it is important that the management of IoT-aware BPs involves modeling (or inclusion) of necessary information related to energy consumption such as access cost (AC), i.e., the energy costs for processing, and communicating over the network and the Quality of Information

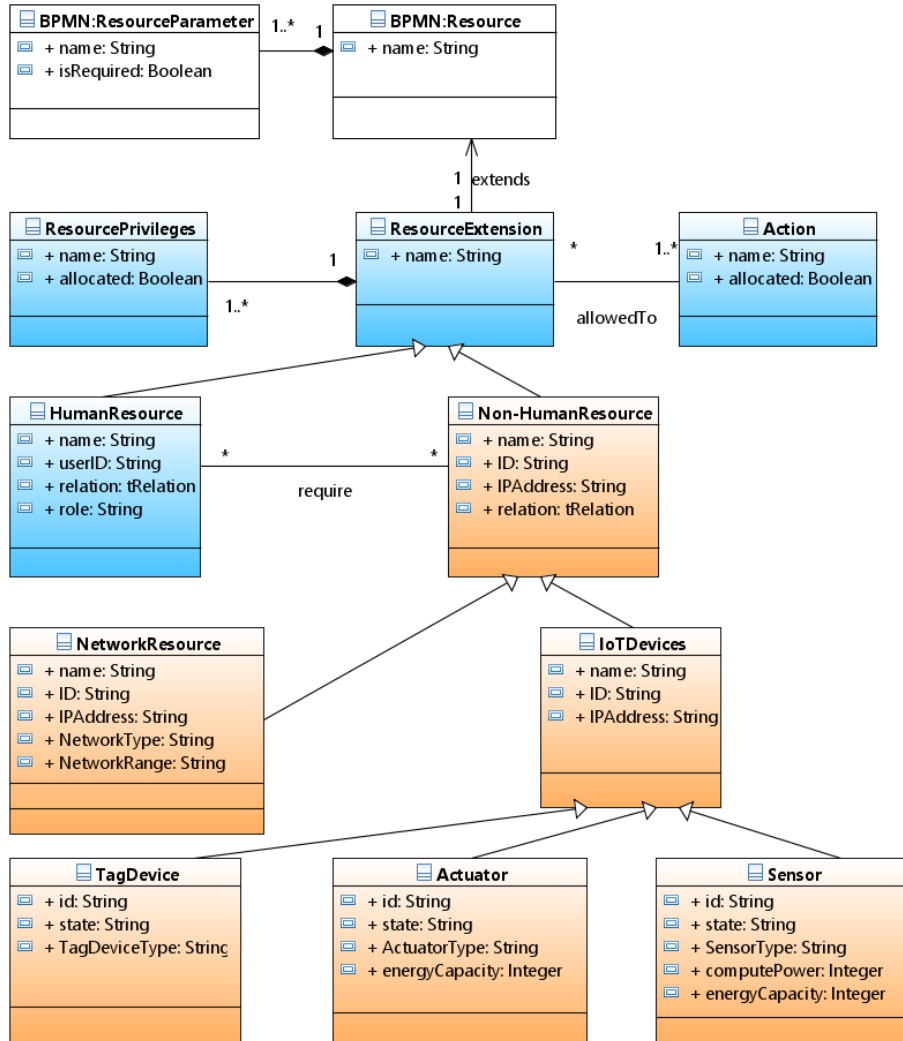


Figure 3.1: Extending resource element in BPMN for IoT devices

(QoI) [39]. In other words, these two criteria are essential for optimal allocation of IoT resources that meet the proposed QoS and SLA requirements of the underlying BPs.

Even though, BPMN 2.0 is one of the most popular modeling notation for modeling BPs there have been only a few works focusing on the importance of gluing (integrating) the IoT domain in BPMN [66, 83]. Recently, a few works [39, 66] have extended BPMN 2.0 to include the QoI [66] and AC [39] features, but the concept of QoI and AC cannot be evaluated interdependent to each other. In fact, the increase in the sampling rate of an IoT device improves the freshness of data, thus providing a higher QoI, but this leads to an increase in energy consumption. Thus, both the QoI and AC must be explicitly included in a process (at design-

time) to keep the QoI and AC optimal during deployment time. Furthermore, changing the QoI and AC requirements in a process will propagate changes to the concrete behavior of a physical resource during the process execution, e.g., reduction in sampling rate (to save battery) will reduce the data quality and can influence the overall quality of monitoring. Thus, we propose to include the energy-aware features such as AC and QoI adopted from Martinho et al. [39] along with their relationships to enable energy-aware modeling of IoT in BPs. These energy-aware concepts are also included in our semantic model, which is detailed in Section 3.3.

3.3 Semantic Formalization of IoT Perspective in BPs

In this section, we describe our cross-domain semantic model that supports the formalization and integration of concepts and relationships from the IoT domain with that of the BPM domain. This semantic model includes various concepts related to the IoT domain such as energy costs, which are needed to formalize the access cost estimations, i.e., computation energy and other costs, along with QoI. These energy-related concepts are important to be modeled during the process design-phase as they affect the overall SLA of a process during the process execution phase. Following the best practices² for ontology development, we developed our semantic model by reusing concepts from existing well-known semantic models. During the initial development of our semantic model [83], we reused the IoT concepts and relationships developed in the IoT-Lite³ ontology [60], which is a part of both the FIWARE project and the FIESTA-IoT ontology [59]. However, in order to keep the IoT-Lite ontology light-weighted and easy to use, it only included the basic concepts from the IoT domain [60, 83]. Being a light-weight ontology, it lacked the coverage of other essential features for IoT devices such as energy cost. Thus, to include and formalize other concepts vital for IoT domain, we reused the concepts from FIESTA-IoT ontology, which being a comprehensive ontology has contributed towards solving interoperability issues in the IoT domain. For the BPM domain, we gave preference to reuse the concepts from the BPMO semantic model [48] rather than using the BPMN 2.0 Ontology⁴ [128]. Even though in this thesis we work mainly with BPMN as the modeling language, we prefer BPMO as it provides generic concepts from the BPM domain that are helpful to attain interoperability with other modeling languages such as EPC. Furthermore, being more generic and easy to use, BPMO has been widely applied in various research projects [48, 72, 73].

²http://wiki.opensemanticframework.org/index.php/Ontology_Best_Practices

³<http://iot.ee.surrey.ac.uk/fiware/ontologies/iot-lite>

⁴<https://dkm.fbk.eu/bpmn-ontology>

3.3.1 Approach Overview

In this section, we provide an overall overview of our approach. As mentioned before, our approach is based on formalizing and integrating the IoT resource perspective in BP models and using semantic technology to enrich these processes. Figure 3.2 illustrates our overall approach, which includes developing a novel comprehensive cross-domain ontology called the *Internet of Things in Business Processes Ontology* (ThingsPrO). More details related to this ontology can be found online at our university webpage^{5,6}. Furthermore, the concepts from this ontology are detailed in Section 3.3.2.

Figure 3.2 illustrates the approach, which consists of three main inputs: (1) IoT-aware process models created using the extended version of BPMN 2.0, (2) IoT Resource description and (3) IoT Resource constraints. The resource constraints comprise of IoT specific rules and properties to be checked for verification and compliance, to ensure optimal management of IoT resources. The ThingsPrO semantic model formally defines the allocated IoT resources that are to be consumed during a process execution along with their specific dependencies. The ThingsPrO ontology is created using RDF language, and the constraints are defined using SWRL formalization. We developed this model using Protégé⁷, which is a popular open-source ontology editor [129]. As illustrated in Figure 3.2, the ThingsPrO ontology is also used to populate a knowledge base that can be easily queried (or reasoned) using SPARQL Protocol and RDF Query Language (SPARQL⁸) queries. This knowledge base will be updated continuously based on the inputs, i.e., re-designing of BP models, or update to properties or constraints. Nevertheless, our contribution is primarily focused on the process design phase of the BPM lifecycle (see Chapter 1) for supporting the development of the IoT-Aware PAIS (or as we call them *Process-and IoT-Aware Information Systems*). Thus, in this work, we make use of the ThingsPrO ontology to support the explicit modeling of the IoT resource perspective in processes by enriching the process models semantically with concepts from the ontology via semantic annotations. Even though this contribution is aimed towards the design-phase, we advocate that our semantic framework and the ontology-based knowledge base is applicable during the process execution phase of the BPM lifecycle. This is because the knowledge base is machine-readable and can be used by any decision-making system to continuously check the IoT specific constraints (resource consumption) during the process execution.

In this contribution, we make use of the structured nature of BPs and integrate the IoT domain into it by including IoT specific concepts (and features)

⁵<http://www-inf.it-sudparis.eu/SIMBAD/tools/IoT-BPO/>

⁶<http://www-inf.it-sudparis.eu/SIMBAD/tools/ThingsPrO/>

⁷<https://protege.stanford.edu/>

⁸<https://www.w3.org/TR/rdf-sparql-query/>

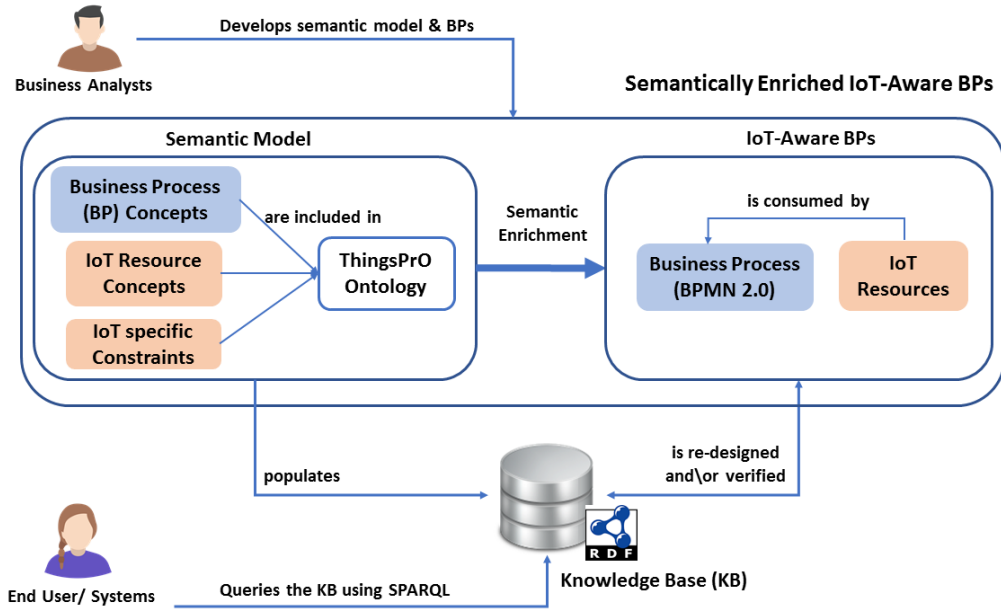


Figure 3.2: Overview of the approach

such as devices, network, energy cost, sensitivity, to name just a few. The concepts from the BP domain, i.e., mainly a task (or activity) is extended to include the resource concept for allocating the IoT concepts that will participate in a process (see Section 3.2.1). Figure 3.3 represents the important concepts and their underlying properties that are present in the ThingsPrO semantic model. The naming convention followed to represent these concepts in Figure 3.3 are as follows: prefix *bpmo* represents the concepts from the BPMO namespace (represented by classes in yellow), the prefix *FiestaIoT* represents the concepts from the Fiesta-IoT namespace (represented by classes in green), and the prefix *thingspro* represents the concepts from the ThingsPrO namespace (represented by classes in blue).

Formally, ThingsPrO is a four-tuple $(C_{ThingsPrO}; A_{ThingsPrO}; R_{ThingsPrO}; S_{ThingsPrO})$, where $C_{ThingsPrO}$ is a set of concepts presented in ThingsPrO ontology (detailed in Section 3.3.2), $A_{ThingsPrO}$ is a set of attributes that belong to the semantic model (detailed in Section 3.3.3), $R_{ThingsPrO}$ is a set of rules defining relations between various concepts (detailed in Section 3.3.4), and $S_{ThingsPrO}$ is a set of strategies to solve the resource allocation conflicts (detailed in Section 3.3.5).

3.3.2 Ontology Concepts

Figure 3.3 represents the concepts in ThingsPrO ontology, wherein the *bpmo : Task* concept is connected to the *thingspro : Resource* concept using the object property *allocate*. There is another object property called *deAllocate* between

the Task concept and the Resource concept. These properties represent the most important relationship between a Task concept reused from the BPMO ontology to a Resource concept developed in our ThingsPrO ontology. Figure 3.3 also depicts other concepts reused from both BPMO and FIESTA-IoT ontologies such as *Process*, *workflowElement*, *IoTDevice* etc.

To clearly differentiate between the two main resources involved in a IoT-aware BP, i.e., human resources and IoT resources, the *thingspro : Resource* is modeled to contain two sub-concepts *thingspro : HumanResource* (representing the humans and their roles) and *thingspro:Non-HumanResource* (representing the IoT device, network, and other enterprise applications). As mentioned earlier, this work is focused on the IoT resource perspective, which is represented via *thingspro : IoTResource* concept, thus we do not go into the details about the human resources. The *FiestaIoT : Device* concept represents the various types of devices in the IoT domain (as per IoT-A reference model [5,66]). This concept is further divided into three sub-types: (1) *FiestaIoT:SensingDevice*, (2) *FiestaIoT:TagDevice* and (3) *FiestaIoT:ActuatingDevice*.

To support optimal IoT resource management and to avoid failures (or conflicts) during the process execution phase, the analysts (model designers) must include the information related to all the IoT specific properties in the process models. In Figure 3.3, it is visible that the *thingspro : Resource* concept is linked to three main concepts that support the management of IoT resources. They are: *thingspro : ResourceType*, *thingspro : ResourcePrivileges* and *thingspro : Constraint* concepts. The *thingspro : ResourceType* concept depicts the type of a resource, wherein a resource can be of one of the two types: (1) logical and (2) physical. Logical, means that the consumption of the resource may lead to a decrease in its availability, e.g., storage space on a device. Physical, means that the consumption of the device does not lead to any decrease in its availability, e.g., the physical part of an RFID Tag device. The concept *thingspro : ResourcePrivileges* signifies that a resource has specific privileges or state regarding the execution of certain actions (allowed or not allowed). The concept *thingspro : Constraint* helps in marking various boundaries while allocating an IoT resource to an activity (detailed in Section 3.3.5.1). For instance, let us take a simple but important constraint, wherein one activity can only be allocated to one type of IoT resource, i.e. the same activity cannot be allocated to a sensor and an actuator. These constraints (simple or complex) enables the process to behaves in the desired manner during the process execution phase. The *thingspro : Constraint* concept is closely related to *thingspro:Conflict* concept that depicts the actual conflicts that might occur during the process execution. These conflicts are handled by resolution strategies represented via *thingspro : Strategy* concept (see Section 3.3.5.2).

For a BP involving an IoT resource (or data), it is crucial to preserve the

right balance between maximizing the QoI and minimizing energy cost. For instance, minimizing the device communication leads to minimization of its energy consumption. For modeling energy cost, we reuse and adapt the concept of AC introduced by Martinho et al. [39]. In Figure 3.3, we relate the concept *thingspro:costParameter* with the concept *FiestaIoT:Device* via object property *hasCost*. It has four sub-classes, they are:

1. *thingspro:energyCost*, which is related with *thingspro:processorEnergyCosts* and *thingspro:radioEnergyCost*, i.e., the costs associated with the central processing units (CPUs) or with activation or maintenance of frequency-related electronic devices (radio emitters/receivers), respectively,
2. *thingspro:communicatioCost*, which depends on the concepts such as, *thingspro : bandwidth*, *thingspro : latency* and *thingspro : radioRange* cost (higher cost for more extended range) that are involved for establishing communication within a specific device,
3. *thingspro:ActuationCost*, which is associated with the cost of performing some actuation task,
4. *thingspro:virtualCost* (not visible in Figure 3.3), which is an abstract class whose concrete value depends on some actual cost (energy or communication cost). However, the resource provider can define any cost.

Furthermore, the *thingspro:costParameter* concept has two object properties: (1) *hasCostValue*, which defines the value of the cost and, (2) *hasCostUnits*, which defines its units.

3.3.2.1 IoT Services & Non-IoT Services

Additionally, it is important to point out the different types of services (or concepts) that are involved when the IoT domain is integrated with the BPM domain. In fact, in context of BPs, there is a major difference between the well studied traditional enterprise services (i.e., web services and Cloud-based services [88, 90, 91, 130, 131]) in context of *Service-Oriented Computing* (SOC) [34] and the IoT services [127]. This is because the IoT devices are highly resource-constrained with respect to energy, computation and storage. This has a direct effect on the expectations from these services or in other words on their QoS [127]. Thus, these services are distinguished in our semantic model using the following concepts:

1. *FiestaIoT : Service*, which is a concept associated with the *FiestaIoT : Device* concept. It represents the functionality of the device exposed via

a service endpoint (resource-constrained service). This service enables a process to access the information about a physical object or entity depicted via *FiestaIoT : Entity* concept.

2. *thingspro : Non – IoTService*, which is a concept that represents the traditional enterprise services, which might also be invoked during the execution of the process.

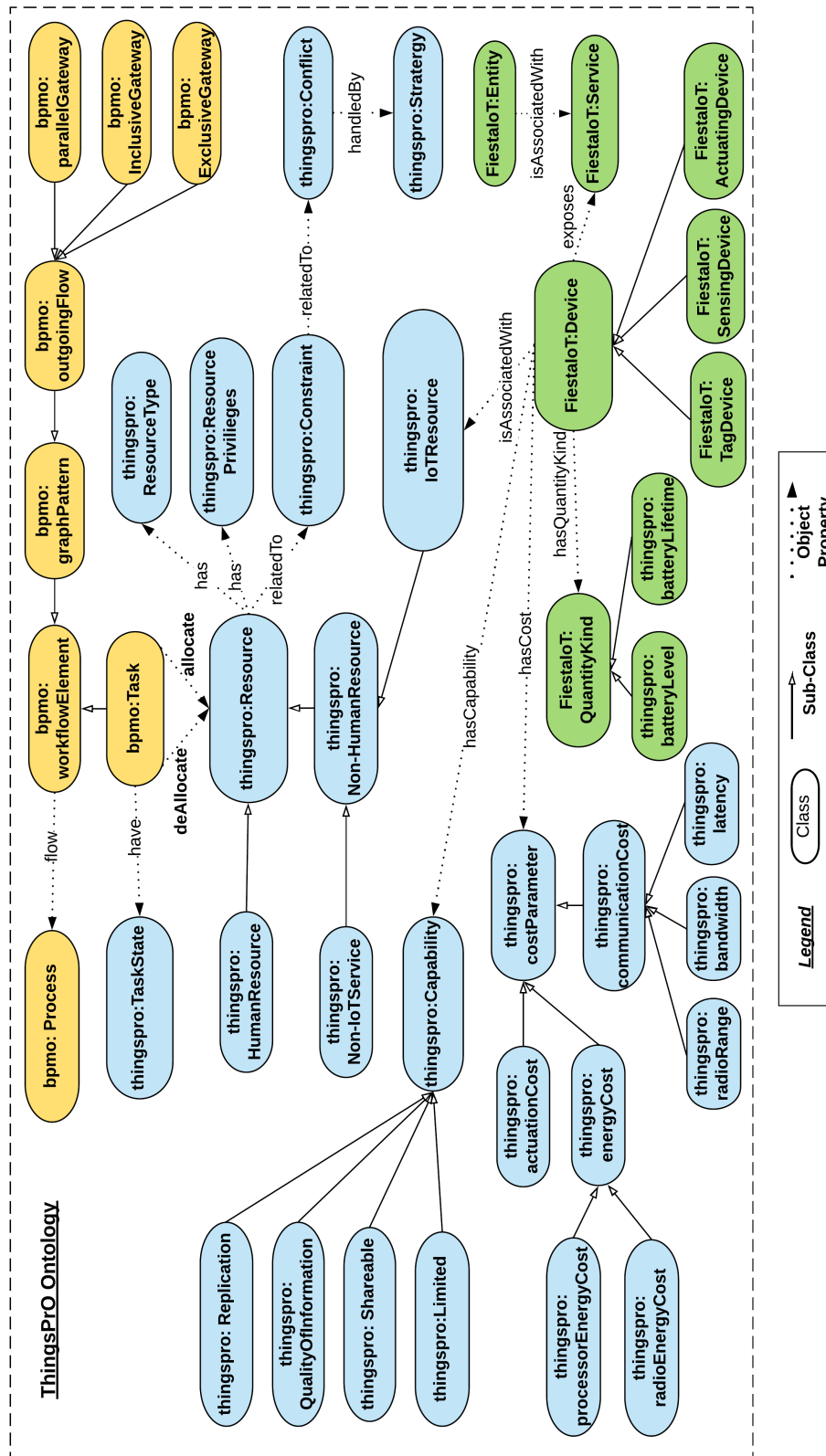


Figure 3.3: Concepts in ThingsPro ontology

3.3.3 Ontology Attributes

In our semantic model, an IoT device is depicted by the attributes where certain attributes are common to all the devices for e.g. IP address, while certain attributes are specific to each device type. For instance, the *FiestaIoT : sensingDevice* concept has specific attributes such as energy cost, response time and latency. Likewise, the *FiestaIoT : actuatingDevice* concept has attributes like speed and force (static force and dynamic force). The *FiestaIoT : TagDevice* concept will have specific attributes such as storage capacity, device type to name a few. These attributes are stored in the tuple represented by $A_{ThingsPro}$.

While modeling the IoT resource allocation in BPs, it is important to keep in mind the various states that a resource and a task go through during their lifecycle [90]. In case of the BP, the concept *bpmo : task* has its current state defined in *thingspro : taskState* having an object property *hasTaskState*. Moreover, a task state may vary among a set of predefined state, i.e., *Initiated*, *Running*, *Canceled*, *Failed* or *Completed* as depicted in Figure 3.4.

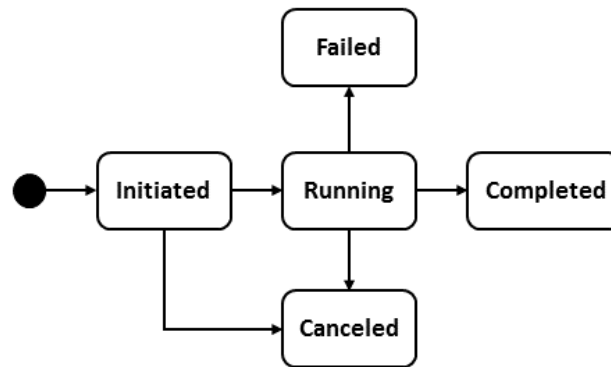


Figure 3.4: Lifecycle depicting states of a task instance

Similarly, an IoT resource will have its set of predefined states in *thingspro : ResourceState* having an object property *hasResourceState*. Thus, the state of an IoT resource may vary from its different states, i.e., *Inactive*, *Allocated*, *Consumed*, *Released* as depicted in Figure 3.5.

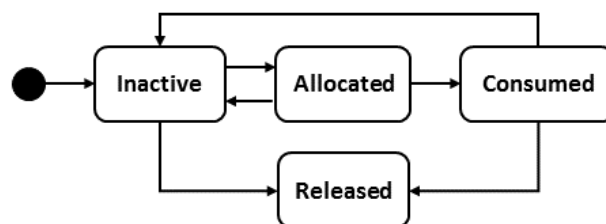


Figure 3.5: Lifecycle depicting states of an IoT resource

Furthermore, each instance of a task goes through these different states during its lifetime. Once a task instance is created, it moves to the *initiated* state. At this state, this instance may be allocated with an IoT resource, this makes the IoT resource to move from *Inactive* to *Allocated* state (see Figure 3.5). Next, an activity instance goes into *Running* state. During the execution of the activity instance, the IoT resource will be under consumption and thus the IoT resource will be in *Consumed* state. From the *Running* state, either the activity instance will be successful and will move to the *Completed* state or will be unsuccessful and will move into *Failed* state. Once the activity instance is over, the IoT resource goes into the *Inactive* state. This resource can be further allocated to other activity instances (based on behavior such as shareability) or can be *Released*. Additionally, an activity instance can go into *Cancelled* state from both *Running* as well as the *Initiated* state.

For instance, in our running process in Figure 1.3 in Chapter 1, when the activity (or task) *a1* consumes an IoT resource, i.e., a *sensor1*, then the *sensor1* moves from *inactive* state to *allocated* and *a1* moves from *initiated* to *running*.

3.3.4 IoT Resource Properties and Relationships in BPs

In this section, we classify the IoT resources on the bases of their specific IoT features, in other words, on the bases of their distinctive properties.

3.3.4.1 IoT Resource Properties in BPs

These IoT specific properties are very relevant during the modeling and development of IoT-aware BPs and must be included in the process models so that they can assist the IT operations teams during the process implementation phase. This is because if ignored then each of these IoT specific features may lead to process failures during the process execution phase. Thus, it is crucial to model all relevant IoT properties during the process design phase.

For instance, an IoT device is associated with a specific energy-related cost parameter represented by *thingspro:costParameter* (see Figure 3.3). Thus, the concept such as the *thingspro:energyCost* and the concept *thingspro:communicationCost* must be modeled as they have impact on the overall SLA of the BP. Furthermore, as visible in Figure 3.3, there are several other characteristics (capabilities) of a IoT device (or the concept *FiestaIoT:Device*) represented via concept *thingspro:capability* and its sub-concepts. They are: (1) *thingspro:shareable*, (2) *thingspro:limited* and (3) *thingspro:QualityOfInformation*.

In other words, it is apt to say that each IoT resource has a set of properties and based on the specific needs of the BPs, these IoT properties must be explicitly defined in the process models. For instance, if the process needs a temperature

sensor with high sensitivity and low power consumption, then this must be included (annotated) in the process models so that during the implementation of the process the operation team is aware of the actual needs. Moreover, these properties are not exclusive for e.g., a resource can be both limited in terms of storage but may be shareable at the same time. In this work, the following IoT specific properties are considered and detailed below. A summary of these properties are provided in Table 3.1.

1. *Access cost* (AC): The cost related to energy usage involved during the processing and communication of data by an IoT device.
 - *High Access Cost* (HAC): The IoT resources in this classification have high energy-related cost such as high power consumption. Some of these devices need to be connected to the main power supply rather than connected to battery power. Such devices tend to have low latency during communication and are *always-on* involving a high energy cost for communication with the network.
 - *Low Access Cost* (LAC): These IoT resources have low energy-related cost and lower power consumption. They are mostly battery operated and the latency involved in such resources is higher with reduced number of communication with the network.
2. *Replication*: Aggregation of the IoT resources (logical or physical) to provide better results in terms of the QoI along with higher confidence in the data.
 - *Horizontal Replication* (HR): The IoT resources to be aggregated must provide a high QoI and high confidence in the data. This can be achieved by mapping multiple physical devices via a logical device or interface. The collection of these devices have a higher sampling rate (i.e., number of observations per unit time). While the collection of these devices may have different capabilities, they must all include the minimum required capability. For instance, a process requiring a high-sensitivity temperature sensor may include a temperature sensor along with a temperature-humidity sensor (both high sensitivity) but not a high-sensitivity light sensor.
 - *Vertical Replication* (VR): The IoT resources to be aggregated may have similar, higher or lower QoI and level of confidence (or reliability) on data. This type of setting is used for including fault-tolerance and to achieve better results than using a single device.
3. *Shareability*: The IoT devices are involved in collaborative cross-organizational BPs. Thus, it is important to explicitly model the shareability feature of the IoT devices (and their data).

- *Shareable* (S): At any given time, an IoT resource (and its data) must be associated and simultaneously used by at least two or more tasks (or activities) from the same process or from different processes.
 - *Non-Shareable* (NS): At any given time, the resource can only be used by one task in a process.
4. *Resource Usage*: In the IoT domain, there are limitations for the use of some devices such as energy or storage limitations that are not in case of other devices.
- *Limited*: In these IoT resources, there exists a maximum limit that has a direct effect on their usage or resource capability such as computation limit, storage limit, battery lifetime. Thus, once the devices reach these limits, the resources are no longer available for usage.
 - *Unlimited*: These IoT resources have no maximum limit for their usage and they can provide a specific capability for longer periods. For instance, a passive RFID Tags (in cards, clothes etc) have no energy limitation as they do not use internal power source.

Furthermore, as mentioned before these IoT properties are crucial to being modeled in the BPs. In case of CPMs (see Chapter 4), the replication and shareability properties are modeled in form of operators included in the CPMs. These operators assist business analysts to configure the CPMs into process models with the desired specifics. Moreover, certain properties may have some contradictions in special cases, for e.g., an RFID device may have unlimited usage as it has no internal power storage, but it is limited in storage. However, in our work, we do not go into the details of such specific cases.

Type of Property		Description
Access Cost	High Access Cost (HAC)	Higher cost related to usage of energy for both computation and communication. Connected to the main power supply
	Low Access Cost (LAC)	Lower cost related to usage of energy for both computation and communication. They are mostly battery operated.
Replication	Horizontal Replication (HR)	Aggregation of IoT resources to provide a higher QoI and confidence.
	Vertical Replication (VR)	Aggregation of IoT resources having similar QoI and confidence parameters.
Shareability	Shareable (S)	IoT devices (and their data) are Shareable between two or more task belonging to the same or different processes.
	Non-Shareable (NS)	IoT devices (and their data) are allocated with only one task belonging to a process.
Limitation of IoT Resource	Limited	Upper limit for usage of IoT resource due to capacity of energy or storage.
	Unlimited	No Upper limit for usage of the IoT resource due to capacity related problem.

Table 3.1: Summary of IoT specific properties for developing IoT-aware BPs

3.3.4.2 Relationships Between IoT and Other Resources in BPs

The semantic model developed in this work provides a cross-domain ontology integrating the IoT domain along with its concepts and its relationships with the BPM domain. At this point, it is important to remember that even though this thesis is entirely focused on IoT resources to develop IoT-aware BPs, we cannot ignore the fact that various other resources such as enterprise web services (in-house or Cloud-based), humans also participate and are consumed alongside IoT resources in the BPs. Thus, it is crucial to manage these IoT resources and formalize the relationships among themselves and also with other resources participating in BPs.

In the ThingsPrO semantic model (see Figure 3.3), the concept *thingspro : allocate* binds a task (or activity) in a process to a resource (human, IoT or non-IoT). Whereas, the concept *thingspro : deallocate* releases a resource from a task, i.e., when a task no more needs to consume a resource it is released. The concept *thingspro:IoTResource* in the Figure 3.3, consists of relationships such as replacement and aggregation, which are possible on this resource. These complex relationships are defined as functions that will be modeled using SWRL formalization and are detailed as follows:

1. Relationship involving only IoT resources:

- *Replacement*: This relationship is represented via the *replacement(r_i, r_j)* function. This function defines that if a IoT device (say r_i) becomes unavailable then there must exist another IoT resource of same type (say r_j) that will replace r_i so that the process continues without failure.
- *Aggregation*: This relationship is represented via function *aggregation(r_i, r_j)*. It defines that two IoT devices of similar kind (say r_i and r_j) are allocated to the same task in a process (via a logical interface). They provide a composition of services (e.g. data), which can have different access cost. This function will implement the replication property of the IoT resources as detailed in Table 3.1.

2. Hybrid relationship involving an IoT resource and an enterprise resource:

- *Integration*: This relationship is represented via function *integration(r_i, s_i)*. It defines the relation between an IoT service and a non-IoT service (detailed in Section 3.3.2.1). In other words, it represents that an IoT service (r_i) is composed or aggregated with a traditional service (non-IoT service s_i). This service composition provides a specific value to the process. For instance, the data from an (IoT) temperature sensor can be integrated with the weather data from a Cloud based service

such as IBM *Weather Company Data*⁹.

3. Hybrid relationship involving an IoT resource and a human resource:

- *Substitution*: This relationship is represented via function *substitution*(h_i, r_i). It represents that a human resource (h_i) can substitute a IoT device (r_i) in case of unavailability. For instance, if a sensor is not available, a human can read the temperature manually and update it in the system. Likewise, if an actuator such as an automated window, in a smart house is not working, a human can perform the specific action.
- *Collaboration*: This relationship is represented via function *collaboration*(h_i, r_i). It represents that both resources, i.e., human (h_i) and IoT resource (r_i) are allocated to the same task to achieve a specific goal (via a logical interface) such as reducing physical stress on a human. For instance, a human helping a robot to move a box. Here, we assume a robot to be a complex IoT resource participating in a production process.

A summary of these complex relationships is presented in Table 3.2.

⁹<https://console.bluemix.net/catalog/services/weather-company-data>

Relation Type	Relation Name	Description
IoT Resources	replacement(r_i, r_j)	r_j replaces r_i , IF r_i becomes <i>unavailable</i>
	aggregation(r_i, r_j)	r_i and r_j are of similar kind and both are allocated to the same task (via logical device). They provide a composition of services (e.g. data).
Hybrid Dependency (Involving Non-IoT Service)	integration(r_i, s_i)	An IoT service r_i is composed with a traditional non-IoT service s_i .
Hybrid Dependency (Involving Human & IoT Resource)	substitution(h_i, r_i)	A human h_i substitutes a device r_i IF the device is unavailable.
	collaboration(h_i, r_i)	A human h_i & a resource r_i are both allocated to the same task to achieve a specific goal.

Table 3.2: Relationships between IoT and other resources

3.3.5 Managing IoT Resource Conflicts using Strategies

The management of the IoT resource allocation in a BP involves the inclusion of constraints on the use of the IoT resources. These constraints restrict the behavior of the resource concerning certain achievable QoS and SLA in the context of the BP. Thus, these constraints are required to be included in the semantic model so that they ensure smooth execution of the process (during the process execution phase of the BPM lifecycle). In our work, we use SWRL formalization to model and express these constraints. Furthermore, in case of a conflict with a resource allocated to a BP, there must be certain strategies in place for resolving these conflicts. In this work, we also provide some of these SWRL based strategies. In Section 3.3.5.1, we detail the constraints on the IoT resources involved in a BP, while in Section 3.3.5.2, we detail the strategies for resolving resource-related conflicts.

3.3.5.1 Constraints on IoT Resources and Task in BPs

In this section, we define the constraints based on information about the IoT specific properties and the relationships that were defined in Section 3.3.4.1 and 3.3.4.2 respectively. These constraints are present in the tuple $R_{ThingsPrO}$ and are categorized into the following: (1) Resource-Property Constraints and (2) Resource-Relationship Constraints.

- *Resource-Property Constraints*: It refers to constraints that are based on the properties specific to IoT devices participating in BPs such as shareability, replication, access cost, to name just a few (see Section 3.3.4.1). The constraints with respect to these IoT specific properties are crucial for the verification of correct IoT resource allocation in the process model (during the process design phase). Thus, based on business needs various resource-specific constraints can be developed by process designers to support better management of the IoT resources in BPs. For instance, we provide an illustration example via Equation 3.1 of one such constraint using SWRL formalism. This constraint depicts a specific property of a sensor device allocated to a task in a BP, wherein the sensor device (*sensor1*) should be *non – shareable*.

Equation 3.1 depicts the aforementioned and is explained as follows: there exists a task (*t1*) at a given time (*time1*) that is allocated to an IoT resource (*sensor1*). This resource (*sensor1*) is available and has a specific property (based on business requirements) that it is *non – shareable*. When the task (*t1*) is in a running state at a time (*time1*), it implies that there cannot be another Task (*t2*) that could be allocated to the same resource (*sensor1*) at

the same time (*time1*). During the execution of the task (*t1*) the resource (*sensor1*) will in a state of *unavailable* for other tasks.

$$\begin{aligned}
& Task(?t1) \wedge Sensor(?sensor1) \wedge hasResState(?sensor1, ?rs) \wedge \\
& \quad swrlb : equal(?rs, "available") \wedge allocate(?t1, ?sensor1) \wedge \\
& hasCapability(?sensor1, ?sh) \wedge swrlb : equal(?sh, "nonshareable") \\
& \quad \wedge hasTaskState(?t1, ?ts) \wedge swrlb : equal(?ts, "running") \Rightarrow \\
& Task(?t2) \wedge Sensor(?sensor1) \wedge hasResState(?sensor1, ?rs) \\
& \quad \wedge swrlb : equal(?rs, "unavailable")
\end{aligned} \tag{3.1}$$

- *Resource-Relationship Constraints*: These type of constraints depict the unique and complex relationships that might exist between different types of IoT or non IoT resources involved in a BP (see Section 3.3.4.2). One of such relationship is given by *thingspro : replacement* concept (see Table 3.2 for other relationships). We provide an illustration example of a constraint related to *thingspro : replacement* concept via Equation 3.2 using SWRL formalism.

Equation 3.2 depicts a constraint existing between two IoT resources (*r1* and *r2*) (both are sensor resources), and both of these sensor resources measure *temperature*. These resources have the same energy costs denoted via *eCost1* and *eCost2*. These resources exhibit the relationship *replacement*, which is given by the function $replacement(r_i, r_j)$ (see Table 3.2). This relationship supports the process during the execution phase, wherein if one sensor resource becomes *unavailable*, then based on the replacement relationship this *unavailable* sensor is replaced by the other sensor as defined in the semantic framework.

$$\begin{aligned}
& Sensor(?r1) \wedge Sensor(?r2) \wedge Quantity(?qType) \\
& \wedge swrlb : equal(?qType; "temperature") \wedge hasQuantityKind(?r1; ?qType) \wedge \\
& \quad hasQuantityKind(?r2; ?qType) \wedge EnergyCost(?eCost1) \\
& \quad \wedge EnergyCost(?eCost2) \wedge hasEnergyCost(?r1; ?eCost1) \wedge \\
& \quad hasEnergyCost(?r2; ?eCost2) \wedge swrlb : equal(?eCost1; ?eCost2;) \Rightarrow \\
& \quad \quad replacement(?r1; ?r2)
\end{aligned} \tag{3.2}$$

3.3.5.2 Strategies for Resolving IoT Resource Conflicts

In this section, we detail the problems arising due to conflicts (or failures) of IoT resources participating in an IoT-aware BP. In such BPs, if there are some conflicts or violation of the constraints formalized in context of the IoT resources

as mentioned in Section 3.3.5.1, then there must be a mechanism to mitigate such failures. In our semantic framework, these conflicts are addressed through a set of strategies that comprises of actions. We specify semantic rules following the (E)vent- (C)ondition-(A)ction structure (On Event If Conditions Do Actions) [132], wherein events represent conflicts and conditions denote constraints. Actions suggest the set of solutions to take for resolving a conflict (i.e., strategy). These strategies are present in the tuple $S_{ThingsPrO}$.

These actions or strategies include replacing an IoT resource such as a sensor by another sensor resource during the execution of a task, or to replace an actuator task with an available human resource. Some examples of the ECA-based conflict resolution strategies are presented in Table 3.3 and are explained in details as follows:

1. Strategy represented as $S_{ThingsPrO1}$ in Table 3.3 in form of SWRL formalism depicts a scenario (i.e. *scenario1*) detailed as follows:
 - Event: Our running example presented in Chapter 1 (see Figure 1.7) represents an IoT-aware BP, wherein a sensor resource *sensor1* is allocated and consumed by task *a1*. Lets assume that this resource becomes *unavailable*.
 - Conditions: If there exists a *replacement* relationship (see Table 3.2) for the *sensor1*, it means that there is a substitute available for it via *sensor2*.
 - Actions: Then *deallocate* the *sensor1* from the task *a1* and allocate the *sensor2* to it.
2. Strategy represented as $S_{ThingsPrO2}$ in Table 3.3 in form of SWRL depicting another solution for the event in *scenario1*:
 - Event: A sensor resource *sensor1*, which is consumed by task *a1*, becomes *unavailable*.
 - Conditions: If there exists a *substitution* relationship (see Table 3.2) for *sensor1*, it means that there is a human substitute available for it via resource *human1*, who can manually perform the sensing task of *sensor1*.
 - Actions: Then *deallocate* the IoT resource *sensor1* from the task *a1* and allocate the human resource *human1* to it.

Furthermore, we are fully aware that there exist several possibilities to define the conflicts and their resolution strategies. It is important to note that these strategies are dependent on the IoT resource properties and their constraints that are modeled by the analysts during the process design phase based on the business needs.

Event/ Conflict	Condition/ Constraint	Action/ SWRL Strategy
<i>S_{ThingsPro1}</i>		
<i>Task(?t1) ∧ Sensor(?Sensor1) ∧ allocate(?t1; ?Sensor1) ∧ hasResState(?Sensor1; ?rs) ∧ swrlb : equal(?rs; "unavailable")</i>	<i>Sensor(?Sensor1) ∧ Sensor(?Sensor1) ∧ Quantity(?qType) ∧ swrlb : equal(?qType; "temperature") ∧ hasQuantityKind(?Sensor1; ?qType) ∧ hasQuantityKind(?Sensor2; ?qType) ∧ replacement(?Sensor1; ?Sensor2)</i>	<i>Task(?t1) ∧ Sensor(?Sensor1) ∧ Sensor(?Sensor2) ∧ deAllocate(?t1; ?Sensor1) ∧ allocate(?t1; ?Sensor2)</i>
<i>S_{ThingsPro2}</i>		
<i>Task(?t1) ∧ Sensor(?Sensor1) ∧ allocate(?t1; ?Sensor1) ∧ hasResState(?Sensor1; ?rs) ∧ swrlb : equal(?rs; "unavailable")</i>	<i>Sensor(?Sensor1) ∧ Human(?Human1) ∧ Quantity(?qType) ∧ swrlb : equal(?qType; "temperature") ∧ hasQuantityKind(?Sensor1; ?qType) ∧ substitution(?Sensor1; ?Human1)</i>	<i>Task(?t1) ∧ Sensor(?Sensor1) ∧ Human(?Human1) ∧ deAllocate(?t1; ?Sensor1) ∧ allocate(?t1; ?Human1)</i>

Table 3.3: Strategies for conflict resolution using SWRL formalism

3.4 Evaluation and Validation

In this section, we present the evaluation and the proof of concept tools for validating our contribution and to demonstrate the feasibility of our work that supports the design of IoT-aware process models enriched with semantic annotations. We also populate an ontology-based knowledge base of heterogeneous process models and heterogeneous IoT resources, which is machine-readable and useful for decision-making purposes.

1. In Section 3.4.1, we provide an evaluation of our proposed ontology developed in Section 3.3 based on the IoT-A reference model [5] using the gold standard method [133]. Next, we compare our ontology to the ontologies from the IoT and BPM domain relevant to our work.
2. In Section 3.4.2, we detail the proof of concept implementation that extends the Signavio¹⁰ process modeling software (open-source version). Signavio supports a web-based application to model BPMN 2.0 processes. We extended Signavio to include the concepts for developing IoT-aware BP models. This extension of the Signavio tool also includes the support for semantic annotations of the IoT-aware BP models with concepts from the ThingsPrO ontology.
3. In Section 3.4.3, we detail the proof of concept implementation for developing an ontology-based knowledge base that contains information about the processes and the IoT resources based on ThingsPrO ontology.

3.4.1 Ontology Validation

To ensure the quality of the content of the ontologies, World Wide Web Consortium (W3C) has published a list of *Best Practice*^{11,12} and *Good Ontologies*¹³. In our work, we follow these best practices and guidelines such as re-using existing semantic models, keeping ontologies simple, keeping the user needs in mind [60] for developing scalable ontologies.

In literature, several evaluation methods exist to ensure the quality of the content of an ontology such as measuring the similarity between ontologies [134], or comparing the ontology to a golden standard [133] (one of the popular approaches). In case of golden standard evaluation technique, various concepts defined in an ontology are checked for its coverage to the classes defined in the gold standard.

¹⁰<https://code.google.com/archive/p/signavio-core-components/source/default/source>

¹¹<https://www.w3.org/TR/swbp-vocab-pub/>

¹²<https://www.w3.org/TR/ldp-bp/>

¹³<https://www.w3.org/wiki/GoodOntologies>

In our work, we compare the IoT-A reference model to check the coverage of the concepts for the IoT domain. We summarize the results in Table 3.4 to show that we covered all-important IoT-A concepts. In our ontology, we cover all the 16 important classes as detailed in the IoT-A reference model and thus attain 100% coverage ratio. Additionally, in order to create a comprehensive cross-domain semantic model, we studied most of the available important ontologies from IoT domain¹⁴ [135]. We found FIESTA-IoT to provide the coverage for most of the concepts that we require to model IoT specific features such as energy concepts from IoT domain. In the context of the BPM domain, we used BPMO ontology and included the concepts available in BPMO ontology to develop our semantic model.

Specifications	Classes	Quantity	Coverage	Ratio
IoT-A	Physical Entity, Device, Actuator, Sensor, Tag, Service, Virtual Entity, Augmented Entity, Resource, Network Resource, On-Device Resource, Digital Artefact, Active Digital Artefact, Passive Digital Artefact, User, Human User	16	16	100%

Table 3.4: Coverage of concepts in IoT-A reference

Furthermore, in Section 2.4, we detailed the various existing semantic models that are relevant to our work. A summary of these semantic models was presented in Table 2.4. In Table 3.5, we compare these existing semantic models with the semantic model developed in this approach. It clearly shows that our approach is comprehensive and covers the concepts from both the domains of BPM and IoT.

¹⁴<http://lov.okfn.org/dataset/lov/vocabs?tag=IoT>

Semantic Models	Sensors	Actuators	Tags	Workflow Elements	Energy Cost	Human Resources
IoT-A	+	+	+	-	-	+/-
FIESTA-IoT	+	+	+	-	+/-	+/-
IoT-Lite	+	+	+	-	-	-
IoT-O [119]	+	+	+	-	-	-
M3 Lite	+	+	+	-	-	-
SSN	+	-	-	-	-	-
SOSA	+	+	-	+	-	-
DogOnt [120]	+	+	+	-	-	-
SAREF	+	+	+	-	-	-
BPMN Ontology	-	-	-	+	-	+
BPMO	-	-	-	+	-	+
Our Approach	+	+	+	+	+	+

Table 3.5: Existing semantic models compared to our approach

3.4.2 Support for Developing IoT-Aware BP Models

To show the feasibility of our work, we extended the Signavio process editor for modeling IoT-aware BPs. This implementation has two layers: (1) to model the IoT-aware BPs (see Section 3.4.2.1), (2) semantic layer implemented to include (and annotate) the concepts from our semantic model (see Section 3.4.2.2).

3.4.2.1 Integrating IoT Resource Definitions in BP Models

As detailed in Section 3.2, the BPMN 2.0 semantics is extended to include different IoT resources, i.e., Sensor Device, Actuator Device and Tag Device, along with network resource. The extended BPMN 2.0 semantics supports the inclusion of the properties of these devices such as energy cost, IP address, accuracy, response time etc.

During the process design-time, analysts can drag and drop the relevant IoT resources from the shape repository into the modeling canvas (i.e., from the extended version of Signavio). Then, these resources are associated with the appropriate task (see area 1 in Figure 3.6). The resource properties can be updated to include the characteristics of the IoT resources. The serialized BPMN 2.0 models (in BPMN 2.0 XML) created using our prototype tool contains both the IoT semantics and the diagram-interchange information about the IoT resources and their association with the activities/tasks in a process model.

3.4.2.2 Semantic Annotations in BP Models

This section details the semantic layer to annotate (associate) a selected activity from the BPMN 2.0 process model with an IoT resource concept, to generate an enriched model as output. IoT resources are described with attributes, properties and their relations (if any exists) as shown in Figure 3.3.

In other words, this tool supports the users to map the relevant IoT concepts provided in ThingsPrO ontology to the activities in the modeling canvas. The ontology is expanded in the tool and can be used easily. To use the ThingsPrO ontology, a user can click and open the button labeled as *Open Ontology File* in the tool (see area 3 in Figure 3.6). Once the button is activated, the same panel will provide all the available concepts defined in the ThingsPrO ontology.

Likewise, the user will be able to select the concept suited to their activity and associate it to them via text annotations [136]. To assist the user in selecting the ontology concepts related to an activity, we provide a recommendation mechanism. Once the user clicks the *ontology* button on the top (see the top of area 1 in Figure 3.6), the system generates a form asking the task to which the resource (ontology concept) needs to be annotated. The system then recommends the

ontology concepts closest to task based on the semantics of its label text. As discussed in various research work [57, 137], such semantic annotation facilitates process modeling without compromising on consumption of time and cognitive load.

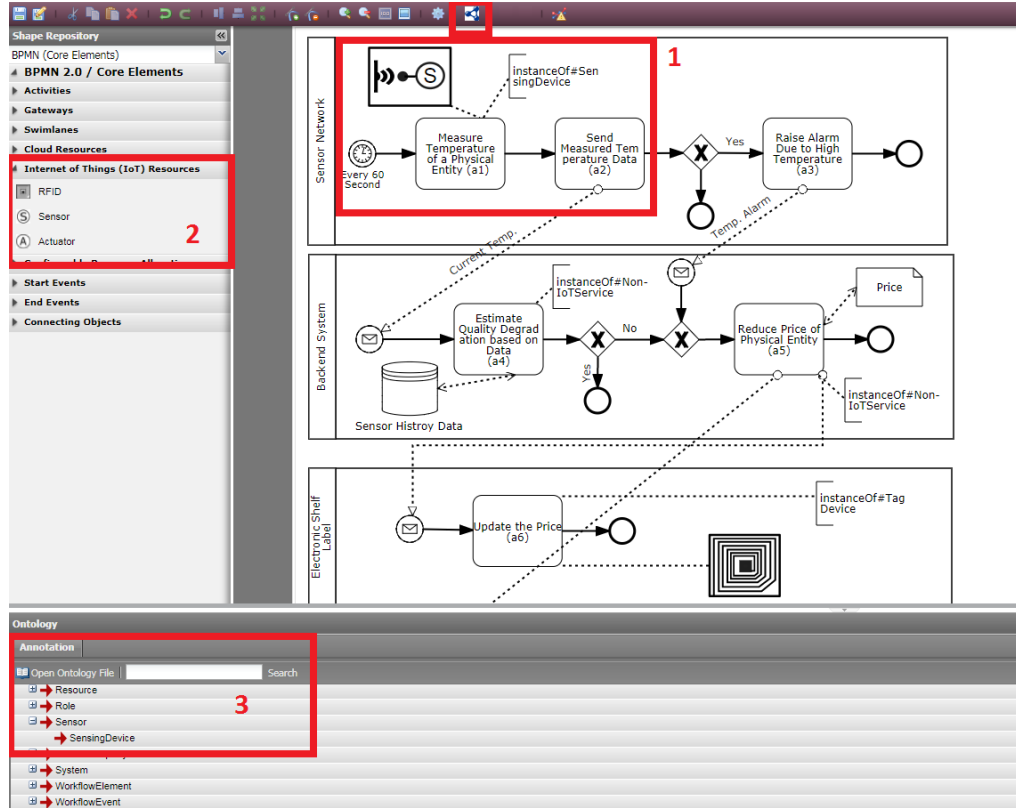


Figure 3.6: Modeling IoT resource and ontology annotation in BPMN 2.0

3.4.3 Ontology-based Knowledge Base

Figure 3.7 represents a prototype implementation to populate an ontology-based knowledge base using the concepts defined in the ThingsPro ontology. This knowledge base (KB) contains the process models enriched with concepts from our semantic model (ThingsPrO). These process models contain the concepts from both the BPM domain along with the IoT domain. Thus, as the process models in this KB are annotated with concepts from ThingsPrO, which reuses the concepts from BPMO ontology, the KB can easily handle processes modeled in different modeling languages (BPMN, EPC etc.). This is because the process modeled in different languages become semantically equivalent at the meta-model level due to the use of generic concepts from BPMO ontology. Furthermore, these models also contain annotation for IoT concepts, making them semantically equivalent

for using the IoT resources (and their data) during the process execution phase. However, in this thesis, we do not go into the details of handling or processing IoT data using our ontology.

Concretely, for creating the ThingsPrO ontology we used Protégé ontology editor [129]. Next, the ThingsPrO ontology is used to build the knowledge base in the form of RDF triples. To use this knowledge base and the triples in a scalable manner, we use an open-source database engine for triplestores from OpenLink Virtuoso¹⁵. The triplestore server can be easily accessed using the Virtuoso Jena Provider API¹⁶. By using the SPARQL endpoint supplied by OpenLink Virtuoso, end users (or system) can easily retrieve and manipulate information using SPARQL queries. This knowledge base contains information about the IoT resources and the process models enriched with the IoT concepts along with the relations between them. It fosters interoperability between processes and underlying heterogeneous IoT resources as the users can define various links and relationships between the resources. Additionally, this knowledge base is queried during the time of conflicts to resolve conflicts based on strategies defined in SWRL rules.

3.4.3.1 Querying the Knowledge Base

Once the shared knowledge base is populated comprising of the IoT resources and the process models, it can be used for various purposes. Being machine-readable, this knowledge base allows checking of the constraints during various phases of BPM lifecycle such as process design or execution phase. It helps to decide and resolve conflicts based on the SWRL ruled defined in the Section 3.3.5.2.

To query the knowledge base, we make use of SPARQL query language. Below, we show a SPARQL query that retrieves a list of IoT devices having an energy cost in *milliwatt*. During a conflict, a decision-making system can execute this query on the knowledge base and use the result to check if the IoT device (mainly sensor) is of correct power usage to optimize the overall energy cost of the process.

Listing 3.1: SPARQL Query on the KB

```
SELECT DISTINCT ?Device ?energyCost ?value
where {?Device ?energyCost ?value ;
rdf:type <http://purl.oclc.org/NET/ssnx/ssn#Sensor>
      FILTER(REGEX(?value,"milliwatt"))}
```

Listing 3.2 depicts the result for SPARQL query represented in Listing 3.1 from the knowledge-base prototype tool. Furthermore, Figure 3.7, provides a screenshot of the aforementioned tool, which is used for creating and querying the ontology-based KB detailed in this section.

¹⁵<http://vos.openlinksw.com/owiki/wiki/VOS/VOSDownload>

¹⁶<http://docs.openlinksw.com/virtuoso/rdfnativestorageprovidersjena/>

Listing 3.2: Result depicting the device description and energy cost

```

<j.5:deviceDescription rdf:datatype="http://www.w3.org/2001/
  ↪ XMLSchema#string">Rugged Wireless Temperature Sensor from
  ↪ RFMicron</j.5:deviceDescription>
<j.1:id>RFM3255</j.1:id>
<j.5:energyCost rdf:datatype="http://www.w3.org/2001/XMLSchema#
  ↪ string">10 milliwatt</j.5:energyCost>
<j.3:accuracy rdf:datatype="http://www.w3.org/2001/XMLSchema#
  ↪ string">80%</j.3:accuracy>

```

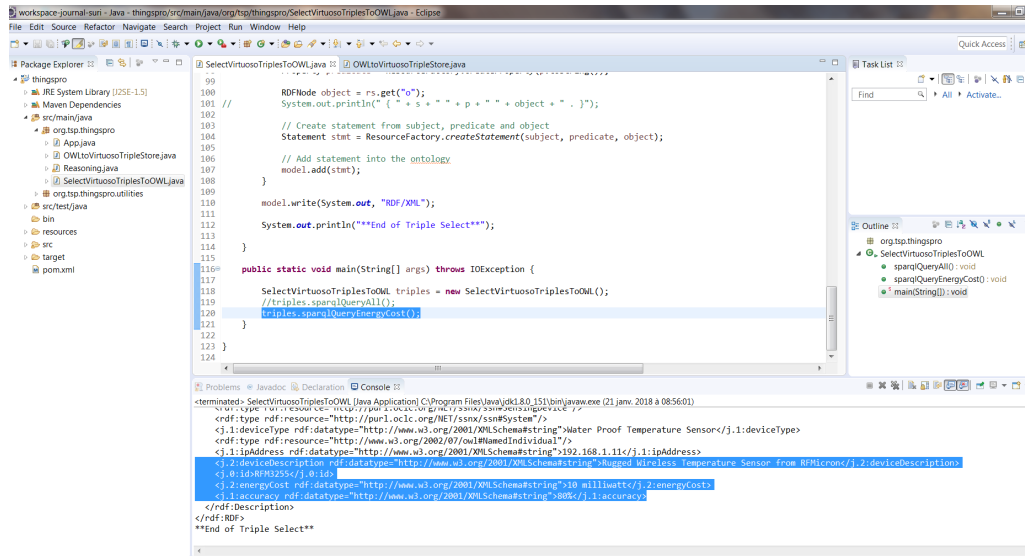


Figure 3.7: Screenshot depicting information retrieval from KB

3.4.4 Threats to Validity

There are some potential threats to the validity of our work, which are detailed as follows:

1. In our work, we have discussed all the three major classes of IoT devices, i.e., Sensors, Actuator and Tags. While in our use cases we have focused more on sensor devices as they are one of the most used IoT resources taking part in the execution of various data-driven PAIS in enterprises globally. However, our contribution can similarly utilize actuators and tag devices. Furthermore, in this work, we do not go into the details of modeling IoT-specific data from the sensor devices.
2. We have shown the feasibility of our work by validation the ontology based on the gold standard technique and by creating proof of concept tools. Yet, this

study calls for the use of a larger dataset involving multiple heterogeneous IoT resources and more process variants to further evaluate the effectiveness of our approach.

3. For the reason of simplicity, the use cases involving the application of IoT and BPM to smart environments, such as Industry 4.0 or Supply Chain 4.0, are kept simple. There are several concepts at the device level or process level that impact a process during its execution and they should be modeled during the design phase such as energy usage based on the type of material monitored, environmental impact etc., which are not taken into account in our current work.
4. We are fully aware that the research on the semantic web and ontology engineering is well developed and we did not investigate into the details of how to collect the knowledge in an effective way or how to efficiently use this knowledge for decision making. Moreover, this work was an initial step towards building an ontology and a KB having elements from both IoT and BPM domain. Thus, we consider the work on further developing and evaluating our approach as a perspective for the future work. More importantly, our work is focused on process modeling and semantic enrichment for integrating the IoT domain into BPM and to provide flexibility to process designers to reuse process know-how by using CPMs as detailed in the Chapter 4.

Nonetheless, as the research on the management of IoT resources in BPM is still in an emerging stage, our comprehensive cross-domain ontology (ThingsPrO) provides a concrete contribution toward integrating concepts from the IoT and BPM domain including relevant concepts such as energy costs.

3.5 Conclusion

In this chapter, we addressed the research problems detailed in Section 1.2.1. They are: how to uniquely integrate the IoT resource perspective in BP models?, How to formally define and include the specific IoT concepts along with their relationships into BPs? and How to formally define heterogeneous IoT resources and their allocations along with the semantically enriched BP models to be shared in a common knowledge base?

To answer these questions, we proposed an extension to BPMN 2.0 meta-model to include the IoT resource perspective along with developing a comprehensive cross-domain semantic model called ThingsPrO. Following the best practices for ontology development, our semantic model capitalizes on existing ontologies, namely FIESTA-IoT and BP MO ontology. It formalizes the concepts and relationships between the IoT domain and BPM domain, along with their constraints. It

also includes strategies to resolve resource-based conflicts, which may arise during the execution phase of BPs. Overall, our contribution fosters error-free allocation of heterogeneous IoT resources and supports interoperability of IoT resources in BPs. Furthermore, this work also supports the enrichment of the process models with essential concepts related to (energy) cost parameters such as access cost. It presents the relationship between these cost parameters and the quality of information w.r.t IoT resources allocated in BPs. Thus, enabling energy-aware management of IoT resources in BPs.

Additionally, our contribution in this thesis assists analysts and process designers to model IoT-aware BP models via a proof of concept tool. This tool enables enrichment of IoT-aware process models with concepts from both IoT and BPM (modeled in ThingsPrO ontology). Likewise, we developed a machine-readable ontology-based knowledge base as another proof of concept tool. This knowledge base contains the information about both the process models and IoT resources (properties and constraints). This knowledge base is useful for decision-making and information retrieval.

Supporting IoT Induced Variability in CPMs

Contents

4.1	Introduction	105
4.2	IoT Resource Perspective in BPs	107
4.3	Configurable IoT-Aware Allocation in CPMs	109
4.3.1	Configurable IoT Assignment Operator	110
4.3.2	Configurable IoT Shareability Operator	112
4.3.3	Configurable IoT Replication Operator	114
4.4	Developing IoT-Aware CPMs: Our Approach vs Classical Approach	116
4.5	Evaluation and Validation	117
4.5.1	Supporting Configurable IoT-Aware Process Modeling	118
4.5.2	Experimentation	119
4.5.3	Threats to Validity	121
4.6	Conclusion	122

4.1 Introduction

This chapter presents the main contribution for this thesis, which is motivated to bridge the research gap for supporting the IoT induced variability in CPMs [8, 63]. Through the approach detailed in this chapter, we address the following research problems: RQ4: How to integrate the variability induced by IoT resources and their features at the CPM level? and RQ5: How to assist the process designers to configure their choices with respect to IoT resources? In other words, the work in this chapter is envisioned to contribute towards expanding the domain of CPMs with capabilities to efficiently manage the IoT resource perspective. This is because CPMs, which is an active area of research for managing process variability in BPM domain [62], facilitates the “Principle of Reuse” in BPs. It supports the modeling and/or (re-)designing the processes by taking into consideration the preexisting knowledge about similar processes (and/or best practices) existing in

an organization, rather than forcing business analysts to design processes “from scratch”. Thus, it facilitates both the flexibility and reuse during the development of BPs, which is necessary to support the rapidly changing business requirements, customer needs or government regulations (in context of smart ecosystems) in context of multi-national organizations spread across the globe.

In the previous chapter (see Chapter 3), based on an extensive literature review, we extended the BP models to include the IoT resource perspective and detailed several concepts from the IoT domain (along with their relationships) that are relevant to the BPM domain. In Section 3.2, we presented an approach to extend the BPMN modeling language for integrating the IoT resources. The work on developing IoT-aware BPs acts as a building block for our work on IoT-aware CPMs. This is because we rely on several concepts defined in Chapter 3 and extend them to include the configuration for managing IoT resources in CPMs. In our work on CPMs, we use the C-BPMN modeling language, which is an extension of the BPMN modeling language (see Section 2.2.1.2). The classical approaches for developing a CPM focus mainly on configuring the control-flow perspective [62], without giving much consideration to the resource perspective. Additionally, in the literature some approaches on CPMs propose the extension of configuration to resources [10, 62, 64, 65], but these approaches are mainly motivated to manage the human resource perspective and are generic to tackle the complexity and specificity involved in the IoT domain (i.e., IoT specific features, constraints, and deployment strategies). Likewise, some recent work proposed the allocation of cloud based computation and storage resources in CPMs [130, 138, 139], but these approaches are not sufficient to model the IoT specific features in CPMs. This is because unlike cloud based resources, IoT devices are highly resource-constrained and exceptionally heterogeneous with respect to specific features such as energy, computation, storage, to name just a few. This has a direct effect on the expectations from these IoT services in term of their QoS along with effect on overall SLA of the BPs (detailed in Section 3.3.2.1). Furthermore, even with a keen interest in research on CPMs, to the best of our knowledge, there has been no uptake to support the integration of the IoT perspective to CPMs. Thus, our work is motivated to address this research gap and to propose solutions for handling variability induced by IoT resources at CPM level.

As mentioned before, in Chapter 3, we detailed several relevant concepts crucial for modeling IoT-aware BPs such as access cost (AC), replication, and shareability (see Section 3.3.4.1). In this chapter, we briefly detail the relevant IoT concepts for supporting variability via CPMs in Section 4.2. Such information is necessary to be modeled in the BPs during the design phase to assist the IT operation teams during the process implementation and execution phase of the BPM life cycle (see Section 1.1). In this work, our approach provides configurable concepts for modeling IoT resource variability, which arises due to aforementioned

specific IoT resource features, i.e., properties and behavior such as shareability and replication, at the CPM level. Concretely, we define a novel approach for developing CPMs with *Configurable IoT Resource Allocation* operators. This allows inclusion of explicit knowledge (options/variability) about various alternatives and constraints that exist for a typical IoT resource based on its behavior relevant to the successful execution of the BP. These IoT-aware CPM can be individualized into a process variant via transformations including both, (1) the control-flow perspective, and (2) IoT resource perspective, to meet a given set of business requirements. Lastly, in context of this work, it is relevant to bring to our notice again that unlike BPs modeled in modeling languages such as BPMN that represent a run-time choice (i.e., BP modeled in BPMN are executable by corresponding process engines), a CPM represents only a design-time choice of an (integrated) family of models, which an analyst will configure (or individualize) for developing a BP variant (based on business needs). Thus, as detailed in our motivating example (see Section 1.3), it is important to model IoT variability at CPM level as a lack of such support hampers the development of IoT-aware BP model variants. This is because there is no reuse of process knowledge, which makes variant creation a time-consuming and error-prone task.

The chapter is organized as follows: Section 4.2, details the importance of modeling IoT resource perspective in CPMs due to the IoT specific features that are needed to be modeled at the CPM level. Section 4.3, describes our approach to model the configurable IoT-aware allocation in CPMs, which enables the explicit modeling of various IoT allocation alternatives in process models. In Section 4.4, we briefly compare the approach proposed in this work with the classical approach for developing CPMs. It provides a clear picture of the advantages of our work for developing IoT-aware CPMs. In Section 4.5, we present the experimentation results associated with our work along with the implemented proof of concept, which clearly demonstrates the feasibility and effectiveness of our contribution. In Section 4.6, we conclude our work.

The contributions presented in this chapter were published in the conference proceeding [140] and received the “Best Student Paper” award.

4.2 IoT Resource Perspective in BPs

In the previous chapter (see Chapter 3), we clearly articulated the need for integrating the IoT and BPM domain for developing processes, especially in relation with the smart environments such as Industry 4.0 [30,117,118,141] including smart Retail/Logistics processes [68]. However, in the context of IoT and BPM, the existing work in the literature [5, 6, 66, 103, 104, 106, 107] have contributed towards modeling and management of IoT resource in BPs at the level of an individual

process model (so-called the BP variant level) rather than developing reusable concepts at the CPM level [37, 83, 84, 101]. Thus, an analyst is forced to design each IoT-aware process model without any guidance (rules or constraints) and support, i.e., process knowledge coming from the CPM level. As a result, these IoT allocations are hard-coded at each individual process in an ad hoc manner, thus making the development of a process variant time-consuming and error-prone. Most importantly, the process enrichment taking place in one process variant is not transmitted to other processes, leading to redundancy and segregation of improvement efforts for each process variant.

Besides, certain features associated with IoT resource perspective leads to significant variability during the design and development of IoT-aware processes (detailed in Section 1.3). It is important to incorporate these features and their resulting behavior at the CPM level. The two key features associated with IoT resources are: (1) replication and (2) shareability. Furthermore, this work assumes that IoT resources constitute both a set of IoT devices and a set of network (e.g., different possible networks provided by Orange Telecom¹ for connection IoT devices), where both are mapped together in a BP model based on the business needs.

Replication has been widely studied for distributed environment because it strongly impacts the following: (1) availability, (2) reliability and (3) performance [142–144]. Reliability and availability have also been widely studied in the context of data-centric services. For instance, Decandia et al. [145] detail their need for creating highly reliable systems and discuss the trade offs between availability, consistency, cost-effectiveness, and performance. Many organizations such as Amazon considers reliability as one of the most significant requirements. This is because the slightest outage can have substantial financial consequences and impacts customer trust [145]. Additionally, each IoT device has a specific access cost (AC) parameter, i.e., device energy consumption cost (processing cost), communication energy cost (cost for bandwidth, latency, and radio range). The access cost and QoI are interdependent as a higher rate of sampling will increase the QoI but will also lead to higher access cost [39] (detailed in Section 3.2.1.1). Basically, in context of IoT (both centralized or distributed architecture), it is essential to explicitly detail and model these replication features to maintain optimal AC and QoI along with high-availability, reliability and fault-tolerance, especially while dealing with time-critical systems (i.e., systems using real-time data for decision making). Furthermore, for the reason of simplicity, in this work, we do not go into the details of modeling each particular concept discussed above separately (i.e., the AC, QoI, availability, and reliability). In our approach, we provide the replication concept that subsumes all these properties. This is because these properties are interrelated, for instance, a large number of devices leads to higher

¹<https://partner.orange.com/orange-iot-networks/>

availability and it helps to mitigate issues related to device failure thus helping to achieve reliability. Also, the QoI is directly influenced by the availability of devices as more number of available devices leads to a higher QoI but it also affects the overall energy cost of the BP.

Likewise, for keeping our approach simple, we consider that our shareability concept subsumes the properties: privacy and security of information, which are highly important in both IoT and BPM domain [146–149]. This is because based on the shareability feature a business can control who can access their data to mitigate the risk of losing control over the data or the tampering of data. We are fully aware of various existing research works that have focused on the topic related to privacy and security in details, however, in this work, we do not go into any such details. Furthermore, the shareability feature is very crucial for the IoT and BPM domain as these IoT devices participating in various BPs capture and transmit data that can contain sensitive or private information such as GPS location, video or audio data. Thus, the BPs must be designed keeping data protection policy in mind (e.g. EU GDPR²). Based on such policies at both the process and IoT resource level, an analyst can design variants having allocated resources that may or may not be shareable between multiple processes or multiple activities of the same process. Overall, this work focuses on modeling and including these IoT resource features at the CPM level based on the approach detailed in Section 4.3, which involves the development of specific configurable IoT resource allocation operators.

4.3 Configurable IoT-Aware Allocation in CPMs

In this section, we detail our approach for supporting the variability due to the inclusion of the IoT resource perspective in BPs and supporting it at the CPM level. This approach involves allocation of IoT resources that takes into account two main parameters: (1) the needed IoT resources and their properties, and (2) the desired resource behavior such as shareability and replication. To support this variability, we identify novel concepts in form of three main operators that support the configuration of resource properties and behavior. These configurable IoT resource allocation operators act as an instrument to model the variability induced by the inclusion of the IoT resource perspective in the CPMs and are described in the following sections. These three operators are:

1. Configurable IoT Assignment operator (A^c) detailed in Section 4.3.1
2. Configurable IoT Shareability operator (S^c) detailed in Section 4.3.2
3. Configurable IoT Replication operator (R^c) detailed in Section 4.3.3

²<https://www.eugdpr.org/>

To illustrate this contribution, we represent a fragment of the process in Figure 4.1, which is taken from the CPM (see Figure 1.5) detailed in our motivating example section in Chapter 1. In this figure, the IoT resources are allocated to activity $a1$ with using the aforementioned configurable resource allocation operators. In the rest of this chapter, we refer to this process fragment in Figure 4.1 to clearly detail the working of these operators. In other words, this figure is revisited and is used to explain the details of each operator along with its functionality. Furthermore, we reuse our motivating example from Chapter 1, wherein Figure 1.7 depicts a BP variant derived from the CPM in Figure 1.5. This, BP variant is enriched with the information about the IoT resource features in form of text annotations. The information included is related to the resource properties and resource behavior. Thus, it depicts one of the many possible BP variants, wherein the activity, for instance, $a1$ is connected to a sensor device and a network resource. Next, we detail our approach to configurable IoT resource allocation via special operators in the following sections.

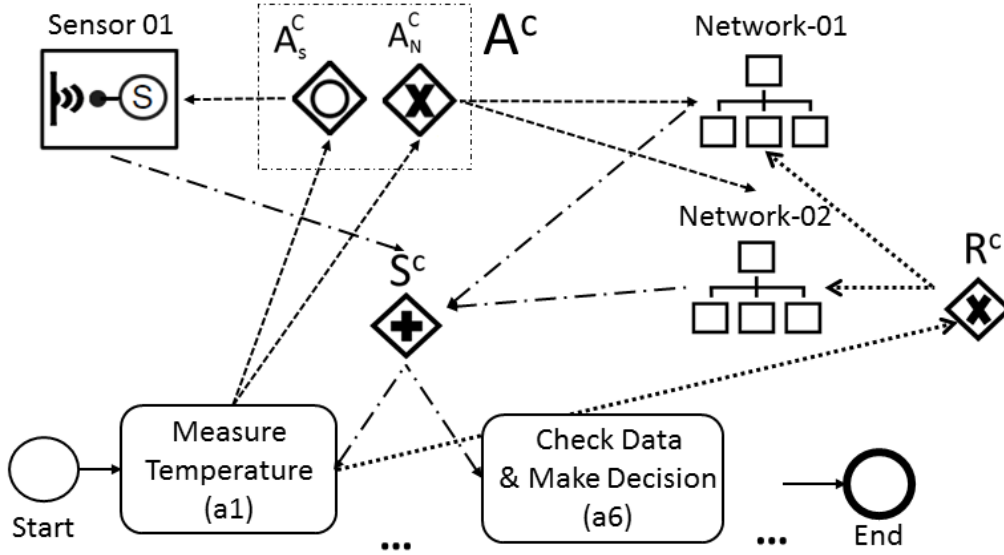


Figure 4.1: Process fragment taken from Figure 1.5 to illustrate configurable IoT allocation operators

4.3.1 Configurable IoT Assignment Operator

The configurable IoT assignment operator (A^c) allows modeling of a variable number of IoT resources allocated to a particular activity. It is the main operator for facilitating the modeling of IoT resource variability at the CPM level. It allows to define a pool of resources and set of guidelines (rules and constraints), which shall be used during the design-time to derive sound process variants [150] with relevant resources allocated to the BP activities. In this work, to illustrate the

different IoT resources that are made up of devices and the underlying network, we categorize them into two main groups: (1) the choice of an IoT device (Sensor, Actuator or Tags), and (2) the choice of a network (based on bandwidth, range and latency). For instance, the process fragment in Figure 4.1, represents an activity to measure temperature ($a1$) that consumes via A^c one sensor (via OR^c gateway associated with A_s^c) and two network resources (via XOR^c gateway associated with A_n^c). It represents a resource variability such as, (1) a temperature sensor and a LoRaWAN based network resource, or (2) a temperature sensor and cellular (2G or 4G) based network resource. Using A^c , CPM can be configured to a specific variant (based on business needs) with assistance from the parameters and guidelines injected in A^c . Further, the A^c operator consists of the following three parameters (summarized in Table 4.1).

Configurable Type: This parameter corresponds to one of the configurable gateways (or connectors), i.e. OR^c , XOR^c , AND^c . These gateways in A^c behave similar to the classical configurable gateway and are configured in the same manner as the connectors detailed in the control-flow perspective of CPMs (see Section 2.2.1.2). They enable the modeling of the allocation behavior of the IoT resource assignment in CPMs. During the design-time, these gateways can change their type in the fashion similar to the classical configurable gateways while preserving the needed behavior and/or restricting the number of allocated resources from the pool of resources. For instance, AND^c is configured to an AND , implying that all devices should be allocated. XOR^c is configured to a XOR , implying that the resource has can be allocated exclusively or cannot be allocated. Whereas OR^c can be configured to AND , OR or XOR , depicting allocation based on the required features of the IoT resource and business needs at the design-time.

Range: This parameter corresponds to the minimum and the maximum number of the resources that can be allocated to an activity, i.e., $rangeD$ for IoT device and $rangeN$ for the network. The range parameter allows imposing a constraint on the choice of configuration by limiting the number of resources that can be allocated with an activity. For instance, the activity $a1$ in Figure 4.1 has a recommendation to include at least one IoT device and one network resources, i.e., $min(rangeD)=1$ and $min(rangeN)=1$. Thus, in this case, the minimum range parameter in the model is set to 1. This will correspond to the allocation shown in process variant in Figure 1.7. The default setting for minimum range equals 0, while maximum range equals the total number of a specific resource allocated to an activity, represented by $|R_D|$ (device) and $|R_N|$ (network).

Assignment Policies: This parameter corresponds to guidelines, i.e., rules and constraints specific to IoT resources for assisting analysts to derive semantically correct process variants. In the current approach, these policies are included in the process models as annotations in form of text or predicate logic in the assignment operator. These annotations support the process designers with relevant informa-

tion needed to model sound IoT-aware process variants. The policies consist of certain default policies along with advanced policies. For instance, (1) an activity should be allocated with an IoT device belonging to only one category, i.e., same activity cannot be allocated to a sensor and to an actuator, (2) an activity can be allocated to multiple resources (e.g., multiple sensors) of the same type or hybrid type, i.e., having at least one of the needed functionality.

Parameters	Behavior and Constraints
Configurable Type (AND^c , OR^c , XOR^c)	Same as classical configurable gateways
Range	$0 \leq range_D \leq R_D $
	$0 \leq range_N \leq R_N $
Assignment Policies	Domain & geography specific constraints

Table 4.1: Parameters for configurable IoT assignment operator

For example, in Figure 4.1, the activity $a1$ can be allocated with a temperature sensor or a hybrid temperature-humidity sensor. Figure 4.1 represents a process fragment (excerpt from Figure 1.5) with an activity $a1$ allocated with one sensors and two network resource. Thus, during the process design phase (design-time), an analyst can configure to keep one sensor and one network resource by transforming the OR^c to AND , and keeping either Network-01 or Network-02 (as represented in the derived process model in Figure 1.7). The AND implies that both the sensor and network are needed. Moreover, such configuration should not violate the range defined for the resources above let us say, $rangeD$ ($min = 1$, $max = 3$); $rangeN$ ($min = 1$, $max = 2$).

4.3.2 Configurable IoT Shareability Operator

Various activities in a BP or different BPs share several IoT resources (and their data). Most of these BPs are collaborative cross-organizational processes that include stakeholders from different organizations or from within the same organization but having different roles and authority (see Section 4.2). Thus, various constraints related to the sharing of the resources and data (based on privacy and security concerns) should account for another layer of variability. For managing this type of variability, we define the configurable IoT shareability operator, represented as S^c . It permits modeling the variability based on the following: (1) the number of different process activities that can share the corresponding IoT resource (i.e., activity-based shareability), and (2) how they share the IoT resources (and their data) within the BP. For the reason of simplicity, in this work, we just keep two possibilities, i.e., shareable or non-shareable. As detailed in Section 4.2, this shareability can include complex behaviors to model privacy and security constraints [147,148], however, its details are out of the scope of this work. This

configurable IoT shareability operator comprises the following three parameters (summarized in Table 4.2). It is to be noted that while using the configurable IoT shareability operator to represent the pool of resources that are shareable or non-shareable, it also implies the use of configurable IoT assignment operator, i.e., $S^c=ON \implies A^c=ON$ (see Figure 4.1). In other words, as the first step the configurable IoT assignment operator is used to model the pool of IoT resources for allocating them to the activities of a CPM and as the next step, the configurable IoT shareability operator is specified to model the shareability behavior between the various activities and the resources.

Configurable Type: this parameter is also similar to classical configurable gateways (i.e., OR^c , AND^c or XOR^c) and is allocated in the same fashion as the configurable IoT assignment operator (see Section 4.3.1). It allows the modeling of the shareability behavior which arises to the IoT specific constraints such as privacy relevant to BPM domain.

Shareability Type: the shareability type parameter ST^c comprises of two types: (1) Shareable (S), and (2) Non-Shareable (NS). Thus, based on the business needs, the ST^c can be configured to one of them in the process model. This will assist the process modelers to make the correct choice of configurations from the CPM while deriving the BP model variants during the design-time.

Shareability Policies: Similar to the assignment policies (see Section 4.3.1), the shareability policies parameter corresponds to guidelines and rules related to privacy and security constraints for the BPs in a specific domain such as health-care, smart transportation, to name just a few. In our approach, these policies are included in the process models as annotations (text or predicate logic) in the shareability operator and support the process designers with relevant information. These policies will be aggregated with the assignment policies during the modeling of the IoT-aware CPMs. For e.g., in a simple healthcare process, the data generated related to the heart condition of a patient via a wearable device should only be shared with the physician and not with their assistant (role-based access control).

Parameters	Behavior and Constraints
Configurable Type (AND^c , OR^c , XOR^c)	Same as classical configurable gateways
Configurable Shareability Type (ST^c)	S , NS
Shareability Policies	Privacy & Security constraints

Table 4.2: Parameters for configurable IoT shareability operator

For instance, to derive a process such as variant-01 (see Figure 1.7) having shareability, the configurable IoT shareability operator in Figure 4.1 must be configured as follows: (1) S^c operator (having AND^c gateway) associated with $a1$,

$a6$, along with the sensor and network resources, and (2) ST^c is configured to a S , to depict data shareability between multiple activities. Further, it is important to note that the replication and shareability operators are semantically dependent on assignment operator. This is because a device needs to be first assigned before it can exhibit replication or shareability behavior. This makes the formal verification for resource allocation an essential work, however, it is out of the scope of this thesis.

4.3.3 Configurable IoT Replication Operator

Each IoT resource allocated to an activity and participating in a BP has specific resource properties such as access cost (AC) (i.e., the energy cost for computing, communication with the network etc.). These properties are relevant to be modeled in a BP at the design-time as they influence the SLA (or other outcomes) of a BP during the run-time. In our approach, we assume that the replication behavior subsumes the aforementioned resource properties (see Section 4.2). Based on the business requirements, the variation in the IoT resource properties leads to the generation of different BP model variants, which behaves in a different way. Thus, we support the modeling of this replication behavior in CPMs via the configurable IoT replication operator (R^c), wherein the resource behavior parameters are expressed in terms of a replication type. The concept of replication rely on the concepts of the previous chapter (see Section 3.3.4.1). These concepts are extended to include configuration concepts via the operator detailed in this section. The replication type has the following two parameters, i.e., Horizontal Replication and Vertical Replication, which are defined as follows:

- Horizontal Replication (HR): This type of replication supports the allocation of multiple choices of resources (devices and network) to an activity to be used concurrently, for instance, consuming the average of the data output (e.g. temperature sensor) from multiple physical devices having the same type (i.e., not combining sensor and actuator) via a logical interface. There are certain constraints associated with this type of replication, they are: (1) all the resources must have the same AC features, and (2) the total number of allocated resources should not exceed the *Range* parameter given by the A^c operator (see Section 4.3.1). Horizontal Replication permits the BP to achieve higher reliability while keeping the energy-related access cost lower. For example, a room having four temperature sensors (all having similar energy costs) connected to an activity via a logical interface, allowing one or more to be active at any given time.
- Vertical Replication (VR): This type of replication also supports the allocation of multiple choices of resources (devices and network) to an activity to

be used concurrently (similar to HR). However, there are no upper or lower bounds to the energy-related access cost of the devices that are to be used together. Similar to HR type, the main constraint associated with this type of replication is that the total number of devices allocated to an activity should not exceed the *Range* parameter given by the A^c operator. The VR type of replication is needed to support high availability and high reliability needed in a BP such as a healthcare process or a supply chain process transporting sensitive goods such as medicines. In other words, as there is no upper bound to the energy-related costs, this replication type is needed for modeling processes used in critical systems. For example, a room fitted with four temperature sensors of different types, i.e., one simple sensor, one hybrid temperature-humidity sensor, and two hybrid temperature-humidity dust resistant sensor, wherein all of them have a different access cost and are mapped via a logical interface.

For modeling the aforementioned variability, the R^c operator has following three parameters that can be set and used by the process designers (summarized in Table 4.3). Similar to the Shareability operator, the use of Replication operator implies the use of assignment operator, i.e., $R^c=ON \implies A^c=ON$

Configurable Type: this parameter depicts the set of resources that can be replicated. The configurable type can be either an OR^c , XOR^c or AND^c (similar to A^c and S^c). During the design-time, a process designer can change the type of these gateways like the classical configurable gateways in CPMs while preserving the needed behavior and/or restricting the number of allocated resources from the pool of resources available for replication. For instance, AND^c is configured to an AND , implying that all devices should be replicated. XOR^c is configured to a XOR , implying that the resource has can be replicated exclusively or some resources will not be involved in the replication based on the requirements from the business stakeholders.

Configurable Replication Type: this parameter depicting the type of replication needed. R^c operator allow the inclusion of various resources in the process model that can be replicated based on the type specified by replication behavior (R^c) parameter, which can be of two types, i.e., HR and VR. Thus, in the model the operator the R^c can be set to one of the possible values, HR or VR.

Replication Policies: similar to the assignment and shareability policies detailed in the above sections the replication policies depict specific guidelines related to the resource properties such as access cost. In our approach, these policies are included in the process models as annotations (text or predicate logic) in the replication operator and support the process designers with relevant information conforming to the domain requirements and SLA. For instance, Figure 4.1 illustrates that both *Network01* or *Network02* can be replicated (let us say with HV),

however as only one of them can be configured at a time, thus they are connected via XOR^c .

Parameters	Behavior and Constraints
Configurable Type (AND^c , OR^c , XOR^c)	Same as classical configurable gateways
Replication Type (R^c)	HR , VR
Replication Policies	Access Cost & QoI related constraints

Table 4.3: Parameters for configurable IoT replication operator

4.4 Developing IoT-Aware CPMs: Our Approach vs Classical Approach

In this section, we provide an illustrative example for developing an IoT-aware CPM based on the approach proposed in this work, i.e., using the configurable IoT resource allocation operators (see Section 4.3). Figure 4.2 represents the allocation of IoT specific features in a process fragment of a CPM. Moreover, it illustrates the difference between the classical approach for developing CPMs and the approach proposed in this thesis. As mentioned earlier, the existing approaches do not consider the IoT resource specificity during the development of a CPM. This forces the duplication of activities in a choice group in order to represent the IoT resource allocation along with representation of IoT specific features such as shareability and replication. Some approaches such as La Rosa et al. [62, 65, 123] support the modeling of resource variability in CPMs. However, they also do not support the modeling of IoT specific features.

In Figure 4.2, the left hand-side depicts the allocation of activity a1 with two sensor devices (one sensor in black and one in orange color) and one network resource. To depict this allocation, the activity a1 is modeled three times and each activity is assigned to an IoT resource. Next, to depict the concept of shareability between activity a1 and a6, the activity a1 and activity a6 are both assigned with one sensor and one network device. This depicts that both the sensor and the network (including its data) are shared between a1 and a6. While another sensor (orange color) is allocated only to the activity a1 to depict the replication behaviour. In fact, it is easy to realize that such modeling of IoT resources in CPMs shall lead to creation of huge process models, which shall be complex to understand and difficult to use. This is because Figure 4.2 represents the allocation of only two sensor and one network, while in reality there may be tens or even hundreds of IoT devices needed in a BP to get the information about the physical world.

To avoid the aforementioned mentioned problem, our proposed approach pro-

vides specialized operators to model IoT resource perspective and their specific features in CPMs. For instance, the right-hand side of Figure 4.2 depicts the modeling of devices and network to the activities using the configurable IoT resource operators. This approach avoids the duplication of activities, which in turn avoids the creation of huge process models that may be complex to understand and difficult to use. Furthermore, our approach provides configuration guidelines to analysts in form of annotations such as text or predicate logic, which assists them during the customization of the CPMs into BP variants. Overall, our approach fosters the development of IoT-aware CPMs by providing easy to use operators and avoiding the creation of complex process models.

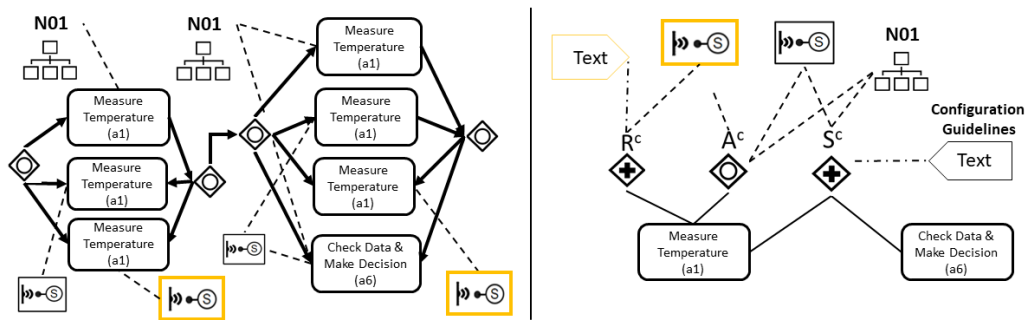


Figure 4.2: Comparison of approaches to develop IoT-aware CPMs

4.5 Evaluation and Validation

In this section, we evaluate and validate our work to show the feasibility and effectiveness of our approach for supporting the inclusion of configurable IoT resources in the CPMs. This section has two main parts as detailed below:

1. In Section 4.5.1, we detail the implementation of a proof of concept tool that supports the modeling of the configurable IoT resource operators. This tool is an extension of an open-source revision of the Signavio editor, which is also used in Chapter 3 to enable the modeling of IoT resources in generic BP models. This extension of Signavio supports the development of a CPM with allocated IoT resources using the configurable IoT resource allocation operators defined in Section 4.3.
2. In Section 4.5.2, we go in-depth on the experimentation performed to evaluate the structural complexity of a process model based on the datasets that were developed using three distinct approaches applied on the same CPM to create the IoT-aware CPMs. Overall, the experimentation result illustrates that our approach reduces the complexity involved in modeling IoT specific features at the CPM level.

4.5.1 Supporting Configurable IoT-Aware Process Modeling

We implemented a proof of concept by extending the Signavio³ process editor (open-source version). Signavio provides a web-based graphical environment for developing process models in BPMN (serializable as BPMN.xml). This extension supports the development of configurable IoT-aware BPs, also detailed in our university web-page⁴. As illustrated in Figure 4.3, our prototype supports the following functionality for managing process variability at design-time:

Modeling the IoT Resource Perspective: As described in the previous chapter, i.e., Chapter 3, we extended the BPMN 2.0 semantics to include concepts from IoT domain, i.e, Sensor, Actuator, RFID and the Network, along with their properties (based on IoT-A framework). These specifications integrated within the Signavio extension allows users to drag and drop IoT resources during the process modeling phase.

Configurable IoT Allocation Operators: These operators assist modeling and integrating the IoT resource perspective at the CPM level by allocating configurable operators to activities based on the approach presented in Section 4.3. These three configurable IoT resource operators, i.e., Assignment (A^c), Shareability (S^c) and Replication (R^c) are used to link the process activities to their allocated IoT resources (e.g., Fig. 4.1). These operators consist of various configurable parameters such as configurable type, configurable replication type, and policies, which will assist the users during development of process variants (design-time).

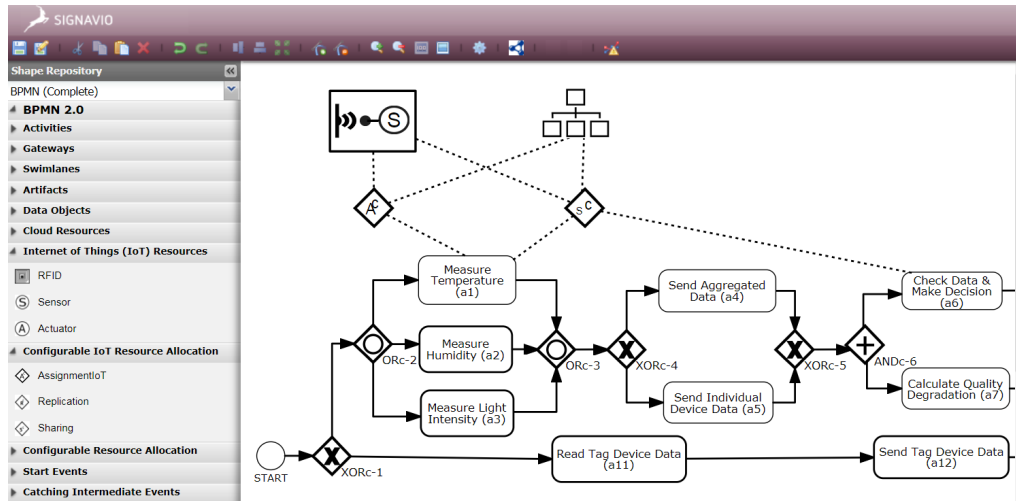


Figure 4.3: Screenshot illustrating the implemented proof of concept

³<https://code.google.com/archive/p/signavio-core-components/source>

⁴<http://www-inf.it-sudparis.eu/SIMBAD/tools/ConfigurableIoTBP>

4.5.2 Experimentation

In this section, we illustrate the effectiveness of our approach by performing experiments on a CPM from the Retail domain as detailed in Figure 1.5 in Chapter 1. This CPM was developed by integrating process variants⁵ adapted from [68]. Our work consolidates both, the control-flow perspective and the IoT resource perspective, along with their allocation strategies for developing configurable IoT-aware process models. Thus, to compare our approach with the current state-of-the-art, we developed the same IoT-aware CPM using three different approaches, detailed as follows:

- First, we develop an IoT-aware CPM using the classical control-flow perspective, which does not consider any variability at the resource level. To do so, an activity is duplicated in the model in a choice block to express the existence of different resource allocation possibilities. For example, this CPM is shown in form of the Process Fragment-1 in Figure 4.4 represents the IoT resource variability and can be individualized based on business requirements. However, it leads to an increase in the structural complexity of the process model.
- Second, we develop an IoT-aware CPM based on the approach from La Rosa et al. [62, 65]. Unlike the classical approach that has no support for modeling the resource perspective, the approach of La Rosa et al. supports the basic resource configuration without any approach to model the complex IoT features such as the resource behavior. Thus, the activities need to be duplicated (similar to the classical approach explained above) to depict these IoT specific features. For instance, an activity may have different shareability requirements in different process variants, which is depicted by duplicating activities and including these features (see Process Fragment-2 in Fig. 4.4), leading to an increase in model complexity.
- Third, we use our approach to develop the IoT-aware CPM, which represents the variability considering both the control-flow perspective and the IoT resource allocation. For instance, the Process Fragment-03 in Figure 4.4 represents our novel approach. In our approach, we can model both the control-flow and the IoT resource perspective in the CPMs, and the overall model complexity is reduced.

Overall, the Figure 4.4 represent three process fragments taken from three separate IoT-aware CPM developed as explained above. For the reason of simplicity, the fragments in Figure 4.4 depict a scenario wherein an activity *a1* is assigned to a *Sensor-01* and a *Network-01*, and both resources are *Shareable*.

⁵<https://github.com/kunalsuri/process-models>

In the classical approach represented via Process Fragment-01, activity $a1$ has been duplicated multiple times to represent the configurable resource assignment concept. One $a1$ is linked to the network resource $N01$ and another $a1$ to $Sensor01$, both $a1$ are connected via a configurable OR . Likewise, to represent the concept of configurable resource shareability between $a1$ and $a6$, the activities $a1$ is duplicated and linked to the IoT resources and connected to $a6$ via a configurable OR . Following the approach in [65] (see Process Fragment-2), the allocation of two IoT resources is done using an OR^c gateway. However, to represent the concept of configurable resource shareability, $a1$ is duplicated and connected with $a6$. Further, based on our approach (see Process Fragment-3), the concept of configurable resource allocation is depicted by linking the IoT resources with activity $a1$ via a configurable IoT assignment operator (A^c). While the resource shareability is represented by linking the resources to activities $a1$ and $a6$ via a configurable IoT shareability operator (S^c).

To evaluate the quality of these three IoT-aware CPMs, we calculate and compare a well-known complexity metric, i.e., Control-Flow Complexity (CFC) [151]. The CFC value related to the XOR gateway is equal to the number of outgoing connectors from that XOR gateway (Equation 4.1). The CFC value for an OR gateway is equal to two raised to the power of the total number of outgoing connectors minus one (see Equation 4.2). While the CFC value of an AND gateway is always one (see Equation 4.3). The total CFC is given by the formula in Equation 4.4, which is the summation of all the CFC values of all the gateways present in a BP model. The CFC_c values assist in evaluating the process complexity in terms of the classical gateways and is used to better understand and examine process models before their actual implementation [151]. As the resource allocation operators are based on the control-flow gateways, we also apply this metric to them. However, we distinguish it by calling it CFC_r .

$$CFC_{XOR-Split}(a) = fan - out(a) \quad (4.1)$$

$$CFC_{OR-Split}(a) = 2^{fan-out(a)-1} \quad (4.2)$$

$$CFC_{AND-Split}(a) = 1 \quad (4.3)$$

$$TotalCFC = \Sigma CFC_{XOR-Split}(a) + \Sigma CFC_{OR-Split}(a) + \Sigma CFC_{AND-Split} \quad (4.4)$$

During our experimentation, we developed three datasets, i.e., one dataset for each approach, wherein each dataset consists of five IoT-aware CPM based on

the same CPM depicted in Figure 1.5 in Chapter 1. These datasets consisted of CPMs developed by allocating IoT resource features with varying complexity.

The results are summarized in Table 4.4. They illustrate that the CPMs modeled using our approach (i.e., dataset 3) have lower aggregated CFC value (i.e., Total $CFC=31$) than other two approaches. Thus, it justifies our hypothesis that the separation in the modeling of the control-flow perspective and the resource perspective leads to a decrease in the overall complexity of the configurable process model. This approach is less prone to errors as the analyst makes use of the configurable IoT resource operators to model the resource variability, rather than duplicating the activities to include such variability in the process models, as seen in the other two approaches. This keeps the models simple and having a less cognitive load on the (humans) business analyst.

As compared to the approach for modeling the CPM in dataset 2 [65], it is visible that the average values for control-flow complexity is lower in our approach (i.e., $CFC_c = 15$) as compared to the other approaches such as dataset 2 having $CFC_c = 30$. While the resource-flow complexity of our approach is higher (i.e., $CFC_r = 16$) than the approach used in dataset 2 (i.e., $CFC_r = 6$). This is because the CPM modeled using our approach supports all the features needed to model the IoT resource variability such as assignment, shareability and replication). Thus, as compared to other approaches, in our approach there is no need to duplicate the activities in the model for representing these resource variability, as a result, it brings down the control-flow complexity. While in the dataset 2, the approach is less expressive and only supports generic resource allocation, thus the values for resource complexity is lower than our approach. As in our approach, the resource variability can be better expressed and thus the number of gateways used is higher, leading to an increase in the CFC_r value.

Complexity Metric	DataSet 1 Classical Approach	DataSet 2 La Rosa Approach	DataSet 3 Our Approach
Average CFC_c	37	30	15
Average CFC_r	N.A.	6	16
Total CFC	37	36	31

Table 4.4: Complexity metrics comparing different approaches

4.5.3 Threats to Validity

There are some potential threats to the validity of our work. Firstly, in our work, we have discussed all the three major classes of IoT devices, i.e., Sensors, Actuator and tags (RFIDs) along with the Network resource. While, the examples are focused more on sensor devices, as they are one of the most used IoT resources

taking part in the execution of various (data-driven) process-oriented information systems in enterprises globally. Nonetheless, our approach can similarly be utilized with actuators and tag devices. However, as pointed out earlier, in this work, we do not go into the details of modeling IoT data concepts in CPMs.

Secondly, as an initial step, we have shown the feasibility of our work by developing and experimenting on a dataset of process variants adapted from [68]. Yet, the study requests for a larger dataset involving multiple process variants and heterogeneous IoT resources (both devices and network possibilities) of varying complexity to further evaluate the effectiveness of our approach.

Third, there is a need to formalize the proposed configurable resource operators and its underlying constraints in order to support the configuration of the CPM and to derive sound process variants. However, this formalization is not covered in the current work and we aim to extend our work to include them in the future.

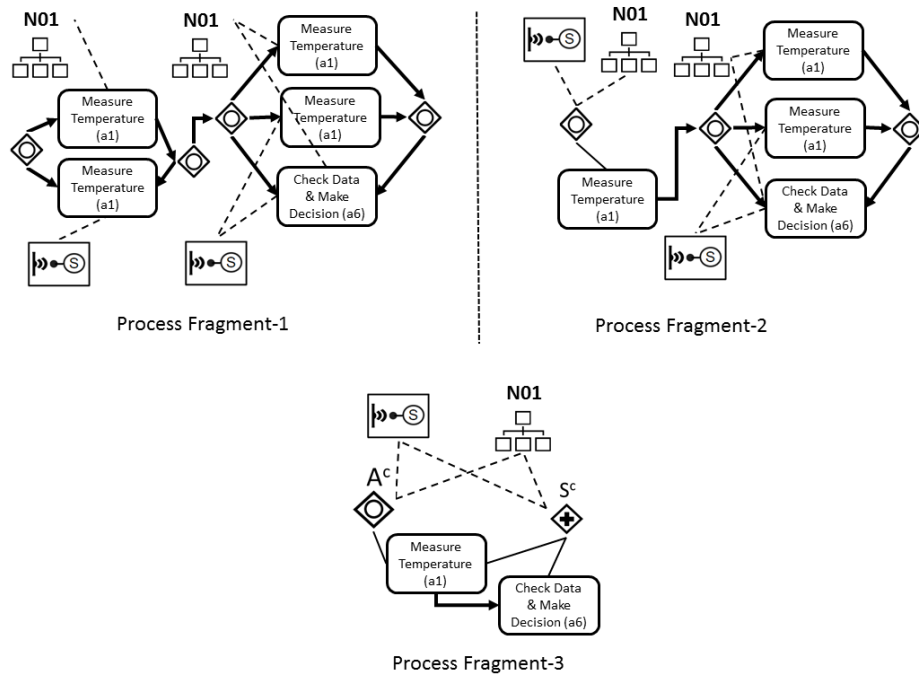


Figure 4.4: Process fragments illustrating three different approaches

4.6 Conclusion

In this chapter, we addressed the two research questions related to our research problems detailed in this thesis (see Section 1.2). These questions are: how to integrate the variability induced by IoT resources and their features at the CPM level? and how to assist the process designers to configure their choices with respect to IoT resources?

To integrate the IoT induced variability, we proposed configuration concepts for IoT resources at the CPM level, wherein we defined a set of configurable IoT-aware allocation operators. These operators support the inclusion of explicit information (options/variability) about various alternatives and constraints for IoT resources. These operators are based on IoT specific features (properties) and behavior such as shareability and replication, which are separate from the control-flow operators.

To support the process designer or business analysts, the IoT resource variability is included in the CPM level. Thus, creating IoT-aware CPMs that can be individualized into a specific process variant via transformations that includes both: (1) the control-flow perspective, and (2) IoT resource perspective, to meet a given set of business requirements. This shifts the knowledge about the management of IoT resource allocation and its customization to the CPM level, wherein an analyst can easily design IoT-aware BP model (variants) based on the guidance (rules and constraints) coming from the CPMs.

Conclusion and Future Works

Contents

5.1 Conclusion	125
5.2 Future Works	127

This thesis work is motivated to answer the research problem detailed in Chapter 1 (see Section 1.2), which is expressed by the following main question : How to support the allocation of IoT resources in configurable business process models? In the previous chapters, we detailed our two contributions that help to answer our research problem. In this chapter, we conclude our work by summarizing these contributions in Section 5.1. Then, in Section 5.2, we present a perspective or outlook in the context of the future research directions of our work.

5.1 Conclusion

The application of PAIS is gaining momentum, especially for large multi-national organizations that have various branches spread across the globe. These organizations need to support different variations of a BP, which come into existence due to the specific needs of the organizations. These needs are based on where these organizations operate and other characteristics such as customer demographics, government regulation, to name just a few. Furthermore, many of these complex BPs have a “physical character” (i.e. interaction with the physical world) and are involved in several business domains such as supply chain and logistics (Industry 4.0), healthcare, smart home automation, to name just a few. These processes rely heavily on the use of heterogeneous devices connected over the internet, forming the IoT (Sensors, Actuators, and Tags), which are orchestrated in a specific sequence to achieve the desired outcome. Thus, there is an evident need for supporting both the variability in BPs (within an organization) and effective allocation (and management) of IoT resources, which is crucial for optimizing costs and achieving a competitive advantage over other organizations. The CPMs assist in variability management of BPs and allow systematic reuse of BPs in a flexible way by sharing the process knowledge among different affiliates

of an organization (i.e. fostering the principle of reuse). These CPMs can be individualized (or customized) into a specific BP variant based on the business requirements. Furthermore, to support the various possible configurations for IoT resources, these CPMs must be enriched to include concepts that can support the IoT resource perspective in CPMs. In other words, the CPMs must support the variability arising from different possible IoT configurations in a BP based on business requirements.

With the growing interest in the integration of the IoT domain with the BPM domain, there has been an increase in research works on this topic in both academia and industry. In the context of the resource perspective in process modeling, various research works have extended the process modeling languages such as BPMN 2.0 with the focus on the human resources involved in a BP. While some research works have proposed approaches to integrate the IoT perspective in BPs, however, these approaches do not consider the heterogeneity of resources (devices and network) in the IoT domain. Further, some existing approaches use semantic technologies to solve the issue of heterogeneity arising due to the use of different process modeling languages such as BPMN, EPC, to name just a few or BPs having text in different natural languages such as English, French and German. Likewise, several projects such as IoT-A, FIWARE, FIESTA-IoT, to name just a few, have developed semantic approaches to mitigate the issues related to heterogeneity in IoT domain due to various devices and their proprietary data (and other) standards. However, to the best of our knowledge, there is no existing work that semantically formalizes and integrates the concepts from the IoT domain to the BPM domain. Furthermore, the approaches for developing CPMs lack the support for integrating the IoT resource perspective in CPMs. In other words, they lack the support for managing the variability in BPs due to different possibilities for selecting IoT resources based on business needs.

Thus, in this thesis, the main objective of our research work is to address the aforementioned challenges by supporting the process designers (analysts) to design IoT-aware BPs and to assist them for including configurations specific to IoT resources in the CPMs. In order to achieve these objectives, we provide two approaches detailed in Chapter 3 and Chapter 4, which enables us to achieve our goal.

In Chapter 3, we detail the first contribution for addressing the problem related to the development of IoT-aware BPs involving heterogeneous IoT resources. This approach is made up of two parts: (1) extending the resource concept of the BPMN 2.0 meta-model with the IoT resource respective based on IoT-A reference model, and (2) by using semantic technologies (ontologies) to model the resource constraints and semantically enriching IoT-aware BPs. Following the best practices for ontology development, we reused the relevant ontologies (and their concepts) from the BPM domain and the IoT domain to create a unified cross-domain se-

semantic model that captures and integrates the concepts and relationships of both these domains. For the same reason, we reused and extended the BPMO ontology (BPM domain) with concepts (classes) from the FIESTA-IoT ontology (IoT domain) to uniformly define heterogeneous IoT resources participating in BPs. This ontology is used to semantically enrich the BP models, which are also stored in a shared common knowledge base. Overall, our semantic framework supports the modeling of IoT-aware BPs by defining IoT specific constraints and relationships w.r.t. BPs. It also provides support to resolve IoT resource-based conflicts using strategies formalized in SWRL.

In Chapter 4, we detail the second contribution for addressing the problem of supporting IoT resource variability in CPMs by extending the CPMs with configurable IoT resource operators. We relied on the IoT concepts and their relationships w.r.t BPs defined in the Chapter 3 and extended them to include configurations. These configurable IoT resource operators support the integration of the IoT resource perspective along with its specific features such as shareability and replication in the context of BPs into CPMs. These configurable operators support the modeling and management of the variability induced by the IoT resources in CPMs and assist process designers to derive different BP variants from CPMs (based on the business needs) in an efficient manner. This approach supports the principle of reuse by allowing the organizations to share their process knowledge along with the information about the underlying IoT resources.

In order to validate our approach, we implemented three proof of concept tools along with the validation of our semantic model. These proof of concepts are implemented by extending the open-source version of Signavio modeler. These tools show the feasibility of our approach and consist of the following: (1) support the modeling of IoT-aware BPs along with their semantic enrichment based on our semantic model as detailed in Chapter 3, (2) application for populating the IoT resource knowledge base also detailed in Chapter 3, and (3) support for modeling configurable IoT-aware resource allocations for designing IoT-aware CPMs as detailed in Chapter 4. Furthermore, we performed experimentation on process models adapted from concrete use cases. Based on the results from these experimentation, the validity and feasibility of our approach are illustrated.

5.2 Future Works

Our research work opens several possible research directions to be achieved in both the short and middle terms. This includes improving the quantity of our work by enriching it with additional IoT properties and constraints along with formalizing them. This will increase the expressive of our approach to support the development of IoT-aware BPs and IoT-aware CPMs. We plan to automate

the configuration approaches to achieve sound process variants from the CPMs and assist a user to find optimal process models with integrated IoT resource perspective based on key performance indicators (KPIs) such as cost, SLA, to name just a few.

Improved Effectiveness: We plan to use a larger dataset involving multiple heterogeneous IoT resources and BP variants to further enrich our contribution that supports the development of IoT-aware BPs using semantic technology. As briefly mentioned in Chapter 3, we plan to develop an approach to include the data-perspective of the IoT resources in our process modeling, which is necessary to explicitly model (during the design-time) the different data sources in IoT-aware BPs such as operational data, historical data, and real-time IoT data.

Semantic Support and Simulations: This thesis work is an initial step towards building an ontology and a Knowledge Base having elements from both IoT and BPM domain. As a future work, we consider investigating the details of how to collect and use the knowledge related to IoT resource in BPs in an effective way for better decision making. This step is needed to support the ontology-based data integration for simulating the IoT-aware BPs by using data from various IoT test beds from different domains such as EU H2020 FIESTA-IoT project. Additionally, we also consider extending the application of our semantic models to include the processes at CPM level.

Automation and Tool Support: We plan to formalize the IoT constraints with respect to the configurable elements such as control-flow or activities and develop algorithms to support automated generation of correct and sound IoT-aware BP models from the IoT-aware CPMs. These automatic techniques are to be implemented in the Sigantio process modeler. Furthermore, to increase the reach of our modeling tool, we plan to implement these approaches in the Eclipse-based framework called Papyrus, which is an industrial grade tool used for model-driven engineering (MDE).

Appendices

List of Publications

Journal Article

1. Kunal Suri, Walid Gaaloul, Arnaud Cuccuru, and Sebastien Gerard. Semantic Framework for Energy-Aware Resource Management of IoT in Business Processes. *International Journal of Systems and Service-Oriented Engineering* (IJSSOE). Volume 8, Issue 1, Article 2. IGI Global, 2018 (DOI: 10.4018/IJSSOE.2018010102).

Conference Proceedings

1. Kunal Suri, Walid Gaaloul, and Arnaud Cuccuru. Configurable IoT-Aware Allocation in Business Processes. *In International Conference on Services Computing*, pp. 119-136. Springer, Cham, 2018 (DOI: 10.1007/978-3-319-94376-3_8). (**Best Student Paper Award**)
2. Kunal Suri, Walid Gaaloul, Arnaud Cuccuru, and Sebastien Gerard. Semantic Framework for Internet of Things-Aware Business Process Development. *In Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), 2017 IEEE 26th International Conference on*, pp. 214-219. IEEE, 2017 (DOI: 10.1109/WETICE.2017.54).
3. Kunal Suri, Juan Cadavid, Mauricio Alferez, Saadia Dhouib, and Sara Tucci-Piergiovanni. Modeling Business Motivation and Underlying Processes for RAMI 4.0-Aligned Cyber-Physical Production Systems. *In Emerging Technologies and Factory Automation (ETF A), 2017 22nd IEEE International Conference on*, pp. 1-6. IEEE, 2017 (DOI:10.1109/ETF A.2017.8247702).
4. Kunal Suri, Arnaud Cuccuru, Juan Cadavid, Sébastien Gérard, Walid Gaaloul, and Samir Tata. Model-based Development of Modular Complex Systems for Accomplishing System Integration for Industry 4.0. *In International Conference on Model-Driven Engineering and Software Development (MODELSWARD)*, pp. 487-495. 2017. (DOI:10.5220/0006210504870495).
5. Mehdi Ahmed-Nacer, Kunal Suri, Mohamed Sellami, and Walid Gaaloul. Simulation of Configurable Resource Allocation for Cloud-Based Business

- Processes. *In Services Computing (SCC), 2017 IEEE International Conference on*, pp. 305-313. IEEE, 2017 (DOI:10.1109/SCC.2017.46).
6. Manoj Kannan S., Kunal Suri, Juan Cadavid, Ion Barosan, Mark Van Den Brand, Mauricio Alferez, and Sebastien Gerard. Towards Industry 4.0: Gap Analysis between Current Automotive MES and Industry Standards using Model-Based Requirement Engineering. *In Software Architecture Workshops (ICSAW), 2017 IEEE International Conference on*, pp. 29-35. IEEE, 2017 (DOI:10.1109/ICSAW.2017.53).

Bibliography

- [1] Marlon Dumas, Marcello La Rosa, Jan Mendling, and Hajo A. Reijers. *Introduction to Business Process Management*, pages 1–33. Springer Berlin Heidelberg, Berlin, Heidelberg, 2018.
- [2] Mathias Weske. Business process management architectures. In *Business Process Management*, pages 333–371. Springer, 2012.
- [3] David Martin, Mark Burstein, Jerry Hobbs, Ora Lassila, Drew McDermott, Sheila McIlraith, Srinu Narayanan, Massimo Paolucci, Bijan Parsia, Terry Payne, et al. Owl-s: Semantic markup for web services. *W3C member submission*, 22(4), 2004.
- [4] Christian Janiesch, Agnes Koschmider, Massimo Mecella, Barbara Weber, Andrea Burattin, Claudio Di Ciccio, Avigdor Gal, Udo Kannengiesser, Felix Mannhardt, Jan Mendling, et al. The internet-of-things meets business process management: mutual benefits and challenges. *arXiv preprint arXiv:1709.03628*, 2017.
- [5] Martin Bauer, Nicola Bui, Jourik De Loof, Carsten Magerkurth, Andreas Nettsträter, Julinda Stefa, and Joachim W. Walewski. *IoT Reference Model*, pages 113–162. Springer Berlin Heidelberg, 2013.
- [6] Sonja Meyer, Andreas Ruppen, and Carsten Magerkurth. Internet of things-aware process modeling: integrating iot devices as business process resources. In *International conference on advanced information systems engineering*, pages 84–98. Springer, 2013.
- [7] R. Seiger, U. Assmann, and S. Huber. A case study for workflow-based automation in the internet of things. In *2018 IEEE International Conference on Software Architecture Companion (ICSA-C)*, pages 11–18, April 2018.
- [8] Michael Rosemann and Wil MP Van der Aalst. A configurable reference modelling language. *Information Systems*, 32(1):1–23, 2007.
- [9] Florian Gottschalk, Wil MP Van Der Aalst, Monique H Jansen-Vullers, and Marcello La Rosa. Configurable workflow models. *International Journal of Cooperative Information Systems*, 17(02):177–221, 2008.
- [10] Alena Hallerbach, Thomas Bauer, and Manfred Reichert. Capturing variability in business process models: the provop approach. *J. of Soft.: Evol. and Process*, 22(6-7):519–546, 2010.
- [11] Marlon Dumas, Marcello La Rosa, Jan Mendling, and Hajo A Reijers. *Fundamentals of Business Process Management*. Springer, 2018.

- [12] Marlon Dumas, Marcello La Rosa, Jan Mendling, and Hajo A. Reijers. *Process-Aware Information Systems*, pages 341–369. Springer Berlin Heidelberg, Berlin, Heidelberg, 2018.
- [13] Marlon Dumas, Wil M. van der Aalst, and Arthur H. ter Hofstede. *Process-aware Information Systems: Bridging People and Software Through Process Technology*. John Wiley & Sons, Inc., New York, NY, USA, 2005.
- [14] Wil M. P. van der Aalst and Kees M. van Hee. *Workflow Management: Models, Methods, and Systems*. MIT Press, 2002.
- [15] Dimitrios Georgakopoulos, Mark Hornick, and Amit Sheth. An overview of workflow management: From process modeling to workflow automation infrastructure. *Distrib. Parallel Databases*, 3(2):119–153, April 1995.
- [16] Wil MP Van Der Aalst, Marcello La Rosa, and Flávia Maria Santoro. *Business process management*, 2016.
- [17] Marlon Dumas, Wil M. Van der Aalst, and Arthur H. ter Hofstede. *Process Aware Information Systems: Bridging People and Software Through Process Technology*. Wiley-Interscience, 2005.
- [18] Jan vom Brocke and Jan Mendling. *Frameworks for Business Process Management: A Taxonomy for Business Process Management Cases*, pages 1–17. Springer International Publishing, Cham, 2018.
- [19] Mirko Kloppenburg, Janina Kettenbohrer, Daniel Beimborn, and Michael Bögle. *Leading 20,000+ Employees with a Process-Oriented Management System: Insights into Process Management at Lufthansa Technik Group*, pages 505–520. Springer International Publishing, Cham, 2018.
- [20] Bartosz Woliński and Saimir Bala. *Comprehensive Business Process Management at Siemens: Implementing Business Process Excellence*, pages 111–124. Springer International Publishing, Cham, 2018.
- [21] Ingo Rau, Iris Rabener, Jürgen Neumann, and Svetlana Bloching. *Managing Environmental Protection Processes via BPM at Deutsche Bahn*, pages 381–396. Springer International Publishing, Cham, 2018.
- [22] Matthias Schrepfer, Matthias Kunze, Gunnar Obst, and Juliane Siegeris. Why are process variants important in process monitoring? the case of zalandose. In *Business Process Management Cases*, pages 431–448. Springer, 2018.
- [23] Object Management Group (OMG). Business process model and notation (bpnm) version 2.0. <https://www.omg.org/spec/BPMN/2.0/>, 2011. Online; accessed 25 Dec. 2018.

- [24] Wil MP Van der Aalst. Formalization and verification of event-driven process chains. *Information and Software technology*, 41(10):639–650, 1999.
- [25] Wil MP van der Aalst. Patterns and xpdL: A critical evaluation of the xml process definition language. *BPM Center report BPM-03-09*, *BPMcenter.org*, pages 1–30, 2003.
- [26] Wil MP Van der Aalst. The application of petri nets to workflow management. *Journal of circuits, systems, and computers*, 8(01):21–66, 1998.
- [27] Object Management Group (OMG). Unified modeling language (uml) version 1.4. <https://www.omg.org/spec/UML/1.4/>, 2001. Online; accessed 25 Dec. 2018.
- [28] Object Management Group (OMG). Unified modeling language (uml) version 2.4.1. <https://www.omg.org/spec/UML/2.4.1/>, 2011. Online; accessed 25 Dec. 2018.
- [29] Paul Grefen, Heiko Ludwig, Samir Tata, Remco Dijkman, Nathalie Baracaldo, Anna Wilbik, and Tim D’Hondt. Complex collaborative physical process management: A position on the trinity of bpm, iot and da. In Luis M. Camarinha-Matos, Hamideh Afsarmanesh, and Yacine Rezgui, editors, *Collaborative Networks of Cognitive Systems*, pages 244–253, Cham, 2018. Springer International Publishing.
- [30] Martin Hankel. Industrie 4.0: The reference architectural model industrie 4.0 (rami 4.0). *ZVEI-German Electrical and Electronic Manufacturers Association*, 2015.
- [31] Luigi Atzori, Antonio Iera, and Giacomo Morabito. The internet of things: A survey. *Computer networks*, 54(15):2787–2805, 2010.
- [32] Frank Leymann, Dieter Roller, and M-T Schmidt. Web services and business process management. *IBM systems Journal*, 41(2):198–211, 2002.
- [33] Organization for the Advancement of Structured Information Standards (OASIS). Web services business process execution language version 2.0. <http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.html>, 2007. Online; accessed 25 Dec. 2018.
- [34] Mike P Papazoglou. Service-oriented computing: Concepts, characteristics and directions. In *Web Information Systems Engineering, 2003. WISE 2003. Proceedings of the Fourth International Conference on*, pages 3–12. IEEE, 2003.

- [35] Wil MP Van der Aalst. Process-aware information systems: Lessons to be learned from process mining. *T. on Petri Nets and Other Models of Conc. II*, 2:1–26, 2009.
- [36] Cristina Cabanillas. Process-and resource-aware information systems. In *Enterprise Distributed Object Computing Conference (EDOC), 2016 IEEE 20th International*, pages 1–10. IEEE, 2016.
- [37] Sonja Meyer, Andreas Ruppen, and Lorenz Hilty. The things of the internet of things in bpmn. In *International Conference on Advanced Information Systems Engineering (CAiSE)*, pages 285–297. Springer, 2015.
- [38] Sylvain Cherrier and Varun Deshpande. From bpm to iot. In *International Conference on Business Process Management*, pages 310–318. Springer, 2017.
- [39] Ricardo Martinho and Dulce Domingos. Quality of information and access cost of iot resources in bpmn processes. *Procedia Technology*, 16:737–744, 2014.
- [40] Luis Jesús Ramón Stroppi, Omar Chiotti, and Pablo David Villarreal. A bpmn 2.0 extension to define the resource perspective of business process models. In *XIV Congreso Iberoamericano en Software Engineering*, 2011.
- [41] Luis Jesús Ramón Stroppi, Omar Chiotti, and Pablo David Villarreal. Extended resource perspective support for bpmn and bpel. In *CIbSE*, pages 56–69, 2012.
- [42] Anastasiia Pika, Moe T Wynn, Colin J Fidge, Arthur HM ter Hofstede, Michael Leyer, and Wil MP van der Aalst. An extensible framework for analysing resource behaviour using event logs. In *International Conference on Advanced Information Systems Engineering*, pages 564–579. Springer, 2014.
- [43] Cristina Cabanillas, Alex Norta, Manuel Resinas, Jan Mendling, and Antonio Ruiz-Cortés. Towards process-aware cross-organizational human resource management. In *Enterprise, Business-Process and Information Systems Modeling*, pages 79–93. Springer, 2014.
- [44] Cristina Cabanillas, David Knuplesch, Manuel Resinas, Manfred Reichert, Jan Mendling, and Antonio Ruiz-Cortés. Ralph: a graphical notation for resource assignments in business processes. In *International Conference on Advanced Information Systems Engineering*, pages 53–68. Springer, 2015.
- [45] Martin Hepp, Frank Leymann, John Domingue, Alexander Wahler, and Dieter Fensel. Semantic business process management: A vision towards

- using semantic web services for business process management. In *e-Business Engineering, 2005. ICEBE 2005. IEEE International Conference on*, pages 535–540. IEEE, 2005.
- [46] Branimir Wetzstein, Zhilei Ma, Agata Filipowska, Monika Kaczmarek, Sami Bhiri, Silvestre Losada, José Manuel López Cobo, and Laurent Cicurel. Semantic business process management: A lifecycle based requirements analysis. In *SBPM*, volume 251, 2007.
- [47] Oliver Thomas and Michael Fellmann. Semantic epc: Enhancing process modeling using ontology languages. *SBPM*, 251, 2007.
- [48] Marin Dimitrov, Alex Simov, Sebastian Stein, and Mihail Konstantinov. A bpmo based semantic business process modelling environment. In *Proceedings of the Workshop on Semantic Business Process and Product Lifecycle Management (SBPM-2007)*, volume 251, pages 1613–0073, 2007.
- [49] Chiara Di Francescomarino, Marco Rospoche, Chiara Ghidini, and Andrea Valerio. The role of semantic annotations in business process modelling. In *Enterprise Distributed Object Computing Conference (EDOC), 2014 IEEE 18th International*, pages 181–189. IEEE, 2014.
- [50] K. Hinkelmann, E. Laurenzi, B. Lammel, S. Kurjakovic, and R. Woitsch. A semantically-enhanced modelling environment for business process as a service. In *2016 4th International Conference on Enterprise Systems (ES)*, pages 143–152, 2016.
- [51] Emanuele Laurenzi, Knut Hinkelmann, Stefano Izzo, Ulrich Reimer, and Alta van der Merwe. Towards an agile and ontology-aided modeling environment for dsml adaptation. In *Advanced Information Systems Engineering Workshops*, pages 222–234. Springer International Publishing, 2018.
- [52] Silvana Pereira Detro, Edoardo Portela, Eduardo Loures Rocha, Hervé Panetto, and Mario Lezoche. Configuring process variants through semantic reasoning in systems engineering. *INSIGHT*, 20(4):36–39, 2017.
- [53] Dennis M Riehle, Sven Jannaber, Patrick Delfmann, Oliver Thomas, and Jörg Becker. Automatically annotating business process models with ontology concepts at design-time. In *International Conference on Conceptual Modeling*, pages 177–186. Springer, 2017.
- [54] Nikolaos Lagos, Adrian Mos, Jean-Yves Vion-Dury, and Jean-Pierre Chanod. Preserving consistency in domain-specific business processes through semantic representation of artefacts. In Witold Abramowicz, editor, *Business Information Systems Workshops*, pages 36–47. Springer International Publishing, 2015.

- [55] Nikolaos Lagos, Adrian Mos, and Mario Cortes-cornax. Towards semantically-aided domain specific business process modeling. *Data Technologies and Applications*, 52(4):463–481, 2018.
- [56] Greta Adamo, Stefano Borgo, Chiara Di Francescomarino, Chiara Ghidini, Nicola Guarino, and Emilio M Sanfilippo. Business processes and their participants: An ontological perspective. In *Conference of the Italian Association for Artificial Intelligence*, pages 215–228. Springer, 2017.
- [57] Jonas Bulegon Gassen, Jan Mendling, Amel Bouzeghoub, Lucinéia Heloisa Thom, and José Palazzo M de Oliveira. An experiment on an ontology-based support approach for process modeling. *Information and Software Technology*, 83:94–115, 2017.
- [58] Matthias Thoma, Torsten Braun, Carsten Magerkurth, and Alexandru-Florian Antonescu. Managing things and services with semantics: A survey. In *Network Operations and Management Symposium (NOMS), 2014 IEEE*, pages 1–5. IEEE, 2014.
- [59] Rachit Agarwal, David Gomez Fernandez, Tarek Elsaleh, Amelie Gyrard, Jorge Lanza, Luis Sanchez, Nikolaos Georgantas, and Valerie Issarny. Unified iot ontology to enable interoperability and federation of testbeds. In *Internet of Things (WF-IoT), 2016 IEEE 3rd World Forum on*, pages 70–75. IEEE, 2016.
- [60] Maria Bermudez-Edo, Tarek Elsaleh, Payam Barnaghi, and Kerry Taylor. Iot-lite: a lightweight semantic model for the internet of things. In *Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBD-Com/IoP/SmartWorld), 2016 Intl IEEE Conferences*, pages 90–97. IEEE, 2016.
- [61] Michael Compton, Payam Barnaghi, Luis Bermudez, Raúl García-Castro, Oscar Corcho, Simon Cox, John Graybeal, Manfred Hauswirth, Cory Henson, Arthur Herzog, et al. The ssn ontology of the w3c semantic sensor network incubator group. *Web semantics: science, services and agents on the World Wide Web*, 17:25–32, 2012.
- [62] Marcello La Rosa, Wil MP Van Der Aalst, Marlon Dumas, and Fredrik P Milani. Business process variability modeling: A survey. *ACM Computing Surveys*, 50(1):2, 2017.
- [63] Florian Gottschalk, Wil MP Van der Aalst, and Monique H Jansen-Vullers. Configurable process models—a foundational approach. In *Reference Modeling*, pages 59–77. Springer, 2007.

- [64] Akhil Kumar and Wen Yao. Design and management of flexible process variants using templates and rules. *Computers in Industry*, 63(2):112–130, 2012.
- [65] Marcello La Rosa, Marlon Dumas, Arthur HM Ter Hofstede, and Jan Mendling. Configurable multi-perspective business process models. *Information Systems*, 36(2):313–340, 2011.
- [66] Sonja Meyer, Klaus Sperner, and Carsten Magerkurth. Towards real world aware enterprise systems-reflecting the quality information of physical resources in services and processes. In *Mobile Adhoc and Sensor Systems (MASS), 2011 IEEE 8th International Conference on*, pages 843–848. IEEE, 2011.
- [67] Kunal Suri and Adrian Mos. Human task monitoring and contextual analysis for domain specific business processes. In *International Conference on Software & Systems Engineering and their Applications (ICSSEA)*, 2015.
- [68] Martin Fiedler and Stefan Meissner. Iot in practice: Examples: Iot in logistics and health. In *Enabling Things to Talk*, pages 27–36. Springer, 2013.
- [69] Object Management Group (OMG). Business process model and notation (bpmn) version 2.0. <https://www.omg.org/spec/BPMN/2.0/>, 2011. Online; accessed 25 Dec. 2018.
- [70] Nour Assy, Walid Gaaloul, and Bruno Defude. Mining configurable process fragments for business process design. In *International Conference on Design Science Research in Information Systems*, pages 209–224. Springer, 2014.
- [71] Marcello La Rosa, Marlon Dumas, Reina Uba, and Remco Dijkman. Business process model merging: An approach to business process consolidation. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 22(2):11, 2013.
- [72] Karn Yongsiriwit, Mohamed Sellami, and Walid Gaaloul. Semantic process fragments matching to assist the development of process variants. In *Services Computing (SCC), 2015 IEEE International Conference on*, pages 712–719. IEEE, 2015.
- [73] Karn Yongsiriwit, Mohamed Sellami, and Walid Gaaloul. A semantic framework supporting business process variability using event logs. In *Services Computing (SCC), 2016 IEEE International Conference on*, pages 163–170. IEEE, 2016.

- [74] Marigianna Skouradaki, Dieter H Roller, Frank Leymann, Vincenzo Ferme, and Cesare Pautasso. On the road to benchmarking bpmn 2.0 workflow engines. In *Proceedings of the 6th ACM/SPEC International Conference on Performance Engineering*, pages 301–304. ACM, 2015.
- [75] Nour Assy, Nguyen Ngoc Chan, and Walid Gaaloul. An automated approach for assisting the design of configurable process models. *IEEE Transactions on Services Computing*, 8(6):874–888, 2015.
- [76] Sara Hachem, Thiago Teixeira, and Valérie Issarny. Ontologies for the internet of things. In *Proceedings of the 8th Middleware Doctoral Symposium*, page 3. ACM, 2011.
- [77] S Meyer and A Ruppen. An approach for a mutual integration of the web of things with business processes. *Enterprise and Organizational Modeling and Simulation*, 153:42–56, 2013.
- [78] Redowan Mahmud, Ramamohanarao Kotagiri, and Rajkumar Buyya. Fog computing: A taxonomy, survey and future directions. In *Internet of Everything*, pages 103–130. Springer, 2018.
- [79] Wil MP Van der Aalst. Formalization and verification of event-driven process chains. *Information and Software technology*, 41(10):639–650, 1999.
- [80] Liliana Cabral, Barry Norton, and John Domingue. The business process modelling ontology. In *Proceedings of the 4th international workshop on semantic business process management*, pages 9–16. ACM, 2009.
- [81] Suparna De, Tarek Elsaleh, Payam Barnaghi, and Stefan Meissner. An internet of things platform for real-world and digital objects. *Scalable Computing: Practice and Experience*, 13(1):45–58, 2012.
- [82] Wei Wang, Suparna De, Ralf Toenjjes, Eike Reetz, and Klaus Moessner. A comprehensive ontology for knowledge representation in the internet of things. In *Trust, Security and Privacy in Computing and Communications (TrustCom), 2012 IEEE 11th International Conference on*, pages 1793–1798. IEEE, 2012.
- [83] Kunal Suri, Walid Gaaloul, Arnaud Cuccuru, and Sebastien Gerard. Semantic framework for Internet of Things-aware business process development. In *26th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)*, pages 214 – 219, Poznan, Poland, June 2017. IEEE.
- [84] Kunal Suri, Walid Gaaloul, Arnaud Cuccuru, and Sébastien Gerard. Semantic Framework for Energy-Aware Resource Management of IoT in Business

- Processes. *International Journal of Systems and Service-Oriented Engineering (IJSSOE)*, 8(1):21 – 43, January 2018.
- [85] Matthias Kloppmann, Dieter Koenig, Frank Leymann, Gerhard Pfau, Alan Rickayzen, Claus von Riegen, Patrick Schmidt, and Ivana Trickovic. Ws-bpel extension for people–bpel4people. *Joint white paper, IBM and SAP*, 183:184, 2005.
- [86] Nick Russell, Wil MP van der Aalst, Arthur HM Ter Hofstede, and David Edmond. Workflow resource patterns: Identification, representation and tool support. In *International Conference on Advanced Information Systems Engineering*, pages 216–232. Springer, 2005.
- [87] Andrea D’Ambrogio and Gregory Zacharewicz. Resource-based modeling and simulation of business processes. In *Proceedings of the Summer Computer Simulation Conference*, page 63. Society for Computer Simulation International, 2016.
- [88] Emna Hachicha and Walid Gaaloul. Towards resource-aware business process development in the cloud. In *Advanced Information Networking and Applications (AINA), 2015 IEEE 29th International Conference on*, pages 761–768. IEEE, 2015.
- [89] Evert F Duipmans, Luis Ferreira Pires, and Luiz O Bonino da Silva Santos. Towards a bpm cloud architecture with data and activity distribution. In *2012 IEEE 16th International Enterprise Distributed Object Computing Conference Workshops*, pages 165–171. IEEE, 2012.
- [90] Mohamed Graiet, Amel Mammam, Souha Boubaker, and Walid Gaaloul. Towards correct cloud resource allocation in business processes. *IEEE Transactions on Services Computing*, 10(1):23–36, 2017.
- [91] Rania Ben Halima, Slim Kallel, Kais Klai, Walid Gaaloul, and Mohamed Jmaiel. Formal verification of time-aware cloud resource allocation in business process. In *OTM Confederated International Conferences” On the Move to Meaningful Internet Systems”*, pages 400–417. Springer, 2016.
- [92] Giray Havur, Cristina Cabanillas, Jan Mendling, and Axel Polleres. Resource allocation with dependencies in business process management systems. In *International Conference on Business Process Management*, pages 3–19. Springer, 2016.
- [93] Nick Russell and Wil MP van der Aalst. Evaluation of the bpel4people and ws-humantask extensions to ws-bpel 2.0 using the workflow resource patterns. *Bpm center report, Department of Technology Management, Eindhoven University of Technology GPO Box*, 513, 2007.

- [94] Luis Jesús Ramón Stroppi, Omar Chiotti, and Pablo David Villarreal. Defining the resource perspective in the development of processes-aware information systems. *Information and Software Technology*, 59:86–108, 2015.
- [95] Cristina Cabanillas, Manuel Resinas, Adela del Río-Ortega, and Antonio Ruiz-Cortés. Specification and automated design-time analysis of the business process human resource perspective. *Information Systems*, 52:55–82, 2015.
- [96] Feng Gao, Maciej Zaremba, Sami Bhiri, and Wassim Derguerch. Extending bpmn 2.0 with sensor and smart device business functions. In *Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), 2011 20th IEEE International Workshops on*, pages 297–302. IEEE, 2011.
- [97] Stephan Haller and Carsten Magerkurth. The real-time enterprise: Iot-enabled business processes. In *IETF IAB Workshop on Interconnecting Smart Objects with the Internet*, pages 1–3. Citeseer, 2011.
- [98] C Timurhan Sungur, Patrik Spiess, Nina Oertel, and Oliver Kopp. Extending bpmn for wireless sensor networks. In *Business Informatics (CBI), 2013 IEEE 15th Conference on*, pages 109–116. IEEE, 2013.
- [99] Jakob Mass, Chii Chang, and Satish Narayana Srirama. Wiseware: A device-to-device-based business process management system for industrial internet of things. In *Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), 2016 IEEE International Conference on*, pages 269–275. IEEE, 2016.
- [100] Robert Wehlitz, Ingo Rößner, and Bogdan Franczyk. Integrating smart devices as business process resources – concept and software prototype. In Lars Braubach, Juan M. Murillo, Nima Kaviani, Manuel Lama, Loli Burgueño, Naouel Moha, and Marc Oriol, editors, *Service-Oriented Computing – IC-SOC 2017 Workshops*, pages 252–257, Cham, 2017. Springer International Publishing.
- [101] Samir Tata, Kais Klai, and Rakesh Jain. Formal model and method to decompose process-aware iot applications. In *OTM Confederated International Conferences” On the Move to Meaningful Internet Systems”*, pages 663–680. Springer, 2017.
- [102] Zakaria Maamar, Thar Baker, Noura Faci, Emir Ugljanin, Yacine Atif, Mohammed Al-Khafajiy, and Mohamed Sellami. Cognitive computing meets the internet of things. In *13th International Conference on Software Technologies*. SciTePress, 2018.

- [103] Zakaria Maamar, Mohamed Sellami, Noura Faci, Emir Ugljanin, and Quan Z Sheng. Storytelling integration of the internet of things into business processes. In *International Conference on Business Process Management*, pages 127–142. Springer, 2018.
- [104] Christian Friedow, Maximilian Völker, and Marcin Hewelt. Integrating iot devices into business processes. In *International Conference on Advanced Information Systems Engineering*, pages 265–277. Springer, 2018.
- [105] Stefan Schönig, Lars Ackermann, and Stefan Jablonski. Internet of things meets BPM: A conceptual integration framework. In *Proceedings of 8th International Conference on Simulation and Modeling Methodologies, Technologies and Applications, SIMULTECH 2018, Porto, Portugal, July 29-31, 2018.*, pages 307–314, 2018.
- [106] Stefan Schönig, Lars Ackermann, Stefan Jablonski, and Andreas Ermer. An integrated architecture for iot-aware business process execution. In *Enterprise, Business-Process and Information Systems Modeling*, pages 19–34. Springer, 2018.
- [107] Uwe Aßmann Ronny Seiger, Peter Heisig. Retrofitting of workflow management systems with self-x capabilities for internet of things. In *International Conference on Business Process Management*. Springer, 2018.
- [108] Lin Ye, Chenglong Hu, Hong-Linh Truong, and Liang Zhang. L2I: Coordinating iot-enabled processes with instant response to rapid changing environments. *ResearchGate preprint*, 2018.
- [109] Sonja Meyer, Klaus Sperner, Carsten Magerkurth, Stefan Debortoli, and Matthias Thoma. Internet of things - architecture, concepts for modelling iot-aware processes. EU FP7 IoT-A project, Deliverable D2. 2, 2012.
- [110] Sankalita Mandal, Marcin Hewelt, and Mathias Weske. A framework for integrating real-world events and business processes in an iot environment. In *OTM Confederated International Conferences” On the Move to Meaningful Internet Systems”*, pages 194–212. Springer, 2017.
- [111] Prosanta Gope and Tzonelih Hwang. Bsn-care: A secure iot-based modern healthcare system using body sensor network. *IEEE Sensors Journal*, 16(5):1368–1376, 2016.
- [112] Carlos Pedrinaci, John Domingue, Christian Brelage, Tammo Van Lessen, Dimka Karastoyanova, and Frank Leymann. Semantic business process management: Scaling up the management of business processes. In *Semantic Computing, 2008 IEEE International Conference on*, pages 546–553. IEEE, 2008.

- [113] Branimir Wetzstein, Zhilei Ma, and Frank Leymann. Towards measuring key performance indicators of semantic business processes. In *International Conference on Business Information Systems*, pages 227–238. Springer, 2008.
- [114] Chiara Di Francescomarino, Chiara Ghidini, Marco Rospocher, Luciano Serafini, and Paolo Tonella. Reasoning on semantically annotated processes. In *International Conference on Service-Oriented Computing*, pages 132–146. Springer, 2008.
- [115] Adrian Mos and Mario Cortes-Cornax. Business matter experts do matter: A model-driven approach for domain specific process design and monitoring. In Marcello La Rosa, Peter Loos, and Oscar Pastor, editors, *Business Process Management Forum*, pages 210–226. Springer International Publishing, 2016.
- [116] Adrian Mos and Mario Cortes-Cornax. Generating domain-specific process studios. In *2016 IEEE 20th International Enterprise Distributed Object Computing Conference (EDOC)*, pages 1–10, 2016.
- [117] S. Manoj Kannan, Kunal Suri, Juan Cadavid, Ion Barosan, Mark Van Den Brand, Mauricio Alférez, and Sébastien Gérard. Towards Industry 4.0: Gap Analysis between Current Automotive MES and Industry Standards Using Model-Based Requirement Engineering. In *IEEE International Conference on Software Architecture Workshops (ICSAW)*, pages 29–35, Gothenburg, Sweden, April 2017. IEEE.
- [118] Kunal Suri, Arnaud Cuccuru, Juan Cadavid, Sebastien Gerard, Walid Gaaloul, and Samir Tata. Model-based development of modular complex systems for accomplishing system integration for industry 4.0. In *5th International Conference on Model-Driven Engineering and Software Development*, pages 487 – 495, Porto, Portugal, February 2017. Scitepress.
- [119] Nicolas Seydoux, Khalil Drira, Nathalie Hernandez, and Thierry Monteil. Iot-o, a core-domain iot ontology to represent connected devices networks. In *European Knowledge Acquisition Workshop*, pages 561–576. Springer, 2016.
- [120] Dario Bonino and Fulvio Corno. Dogont-ontology modeling for intelligent domotic environments. In *International Semantic Web Conference*, pages 790–803. Springer, 2008.
- [121] Wil MP van der Aalst, Alexander Dreiling, Florian Gottschalk, Michael Rosemann, and Monique H Jansen-Vullers. Configurable process models as a basis for reference modeling. In *International Conference on Business Process Management*, pages 512–518. Springer, 2005.

- [122] Arnd Schnieders and Frank Puhmann. Variability mechanisms in e-business process families. *BIS*, 85:583–601, 2006.
- [123] Marcello La Rosa, Marlon Dumas, Arthur HM ter Hofstede, Jan Mendling, and Florian Gottschalk. Beyond control-flow: Extending business process configuration to roles and objects. In *International Conference on Conceptual Modeling*, pages 199–215. Springer, 2008.
- [124] Andreas Ruppen and Sonja Meyer. An approach for a mutual integration of the web of things with business processes. In *Workshop on Enterprise and Organizational Modeling and Simulation*, pages 42–56. Springer, 2013.
- [125] Suparna De, Yuchao Zhou, and Klaus Moessner. Ontologies and context modeling for the web of things. In *Managing the Web of Things*, pages 3–36. Elsevier, 2017.
- [126] Klaus Sperner, Sonja Meyer, and Carsten Magerkurth. Introducing entity-based concepts to business process modeling. In *International Workshop on Business Process Modeling Notation*, pages 166–171. Springer, 2011.
- [127] Matthias Thoma, Sonja Meyer, Klaus Sperner, Stefan Meissner, and Torsten Braun. On iot-services: Survey, classification and enterprise integration. In *Green Computing and Communications (GreenCom), 2012 IEEE International Conference on*, pages 257–260. IEEE, 2012.
- [128] Marco Rospocher, Chiara Ghidini, and Luciano Serafini. An ontology for the business process modelling notation. In *FOIS*, pages 133–146, 2014.
- [129] Mark A Musen. The protégé project: a look back and a look forward. *AI matters*, 1(4):4–12, 2015.
- [130] Mehdi Ahmed-Nacer, Kunal Suri, Mohamed Sellami, and Walid Gaaloul. Simulation of configurable resource allocation for cloud-based business processes. In *14th International Conference on Services Computing (SCC)*, pages 305 – 313, Honolulu, United States, June 2017. IEEE.
- [131] Souha Boubaker, Walid Gaaloul, Mohamed Graiet, and Nejib Ben Hadj-Alouane. Event-b based approach for verifying cloud resource allocation in business process. In *2015 IEEE International Conference on Services Computing (ICSOC)*, pages 538–545. IEEE, 2015.
- [132] George Papamarkos, Alexandra Poulouvasilis, and Peter T Wood. Event-condition-action rule languages for the semantic web. In *Proceedings of the First International Conference on Semantic Web and Databases*, pages 294–312. Citeseer, 2003.

- [133] Marta Sabou and Miriam Fernandez. Ontology (network) evaluation. In *Ontology engineering in a networked world*, pages 193–212. Springer, 2012.
- [134] Alexander Maedche and Steffen Staab. Measuring similarity between ontologies. In *International Conference on Knowledge Engineering and Knowledge Management*, pages 251–263. Springer, 2002.
- [135] Garvita Bajaj, Rachit Agarwal, Pushpendra Singh, Nikolaos Georgantas, and Valerie Issarny. A study of existing ontologies in the iot-domain. *arXiv preprint arXiv:1707.00112*, 2017.
- [136] Chiara Ghidini, Chiara Di Francescomarino, Marco Rospocher, Paolo Tonella, and Luciano Serafini. Semantics-based aspect-oriented management of exceptional flows in business processes. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(1):25–37, 2012.
- [137] Frederik Gailly, Sven Casteleyn, and Nadejda Alkhaldi. On the symbiosis between enterprise modelling and ontology engineering. In *International Conference on Conceptual Modeling*, pages 487–494. Springer, 2013.
- [138] Emna Hachicha, Nour Assy, Walid Gaaloul, and Jan Mendling. A configurable resource allocation for multi-tenant process development in the cloud. In *Advanced Information Systems Engineering (CAiSE)*, pages 558–574. Springer, 2016.
- [139] Molka Rekik, Khouloud Boukadi, Nour Assy, Walid Gaaloul, and Hanène Ben-Abdallah. A linear program for optimal configurable business processes deployment into cloud federation. In *Services Computing (SCC), 2016 IEEE International Conference on*, pages 34–41. IEEE, 2016.
- [140] Kunal Suri, Walid Gaaloul, and Arnaud Cuccuru. Configurable IoT-Aware Allocation in Business Processes. In *International Conference on Services Computing (SCC)*, pages 119–136, Seattle, USA, 2018. Springer.
- [141] Kunal Suri, Juan Cadavid, Mauricio Alférez, Saadia Dhouib, and Sara Tucci-Piergiovanni. Modeling business motivation and underlying processes for RAMI 4.0-aligned cyber-physical production systems. In *22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, pages 1–6, Limassol, France, September 2017. IEEE.
- [142] Ouri Wolfson, Sushil Jajodia, and Yixiu Huang. An adaptive data replication algorithm. *ACM Transactions on Database Systems (TODS)*, 22(2):255–314, 1997.

- [143] Sushant Goel and Rajkumar Buyya. Data replication strategies in wide-area distributed systems. In *Enterprise service computing: from concept to deployment*, pages 211–241. IGI Global, 2007.
- [144] Waleed Bin Qaim and Ozgur Ozkasap. Draw: Data replication for enhanced data availability in iot-based sensor systems. In *2018 IEEE 16th Intl Conf on Dependable, Autonomic and Secure Computing, 16th Intl Conf on Pervasive Intelligence and Computing, 4th Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech)*, pages 770–775. IEEE, 2018.
- [145] Giuseppe DeCandia, Deniz Hastorun, Madan Jampani, Gunavardhan Kukulapati, Avinash Lakshman, Alex Pilchin, Swaminathan Sivasubramanian, Peter Voshall, and Werner Vogels. Dynamo: amazon’s highly available key-value store. In *ACM SIGOPS operating systems review*, volume 41, pages 205–220. ACM, 2007.
- [146] Adam Barth, John Mitchell, Anupam Datta, and Sharada Sundaram. Privacy and utility in business processes. In *20th IEEE Computer Security Foundations Symposium (CSF’07)*, pages 279–294. IEEE, 2007.
- [147] Faiza Loukil, Chirine Ghedira-Guegan, Aïcha Nabila Benharkat, Khoulood Boukadi, and Zakaria Maamar. Privacy-aware in the iot applications: a systematic literature review. In *OTM Confederated International Conferences” On the Move to Meaningful Internet Systems”*, pages 552–569. Springer, 2017.
- [148] Mahmoud Barhamgi, Charith Perera, Chirine Ghedira, and Djamel Benslimane. User-centric privacy engineering for the internet of things. *IEEE Cloud Computing*, 5(5):47–57, 2018.
- [149] Jan Mendling, Ingo Weber, Wil Van Der Aalst, Jan Vom Brocke, Cristina Cabanillas, and et al. Blockchains for business process management - challenges and opportunities. *ACM Transactions on Management Information Systems (TMIS)*, 9(1):4:1–4:16, February 2018.
- [150] Wil MP Van Der Aalst, Kees M Van Hee, Arthur HM Ter Hofstede, Natalia Sidorova, HMW Verbeek, Marc Voorhoeve, and Moe Thandar Wynn. Soundness of workflow nets: classification, decidability, and analysis. *Formal Aspects of Computing*, 23(3):333–363, 2011.
- [151] Jorge Cardoso. Evaluating the process control-flow complexity measure. In *Web Services, 2005. ICWS 2005. Proceedings. 2005 IEEE International Conference on*. IEEE, 2005.

