The Things of the Internet of Things in BPMN

Sonja Meyer[1], Andreas Ruppen[2], Lorenz Hilty[3]

[1] Informatics and Sustainability Research Group, Swiss Federal Laboratories for Materials
Science and Technology, 9014 St. Gallen, Switzerland,
sonja.meyer@empa.ch
[2] Software Engineering Group, University of Fribourg, 1700 Fribourg, Switzerland,
andreas.ruppen@unifr.ch
[3] Department of Informatics, University of Zurich, 8050 Zurich, Switzerland
hilty@ifi.uzh.ch

Abstract. The component "thing" of the Internet of Things does not yet exist in current business process modeling standards. The "thing" is the essential and central concept of the Internet of Things, and without its consideration we will not be able to model the business processes of the future, which will be able to measure or change states of objects in our real-world environment. The presented approach focuses on integrating the concept of the Internet of Things into the meta-model of the process modeling standard BPMN 2.0 as standard-conform as possible. By a terminological and conceptual delimitation, three components of the standard are examined and compared towards a possible expansion. By implementing the most appropriate solution, the new thing concept becomes usable for modelers, both as a graphical and machine-readable element.

Keywords: Internet of Things, BPMN, Physical Entity, Entity of Interest

1 Introduction

Due to the ongoing development of Web technologies the giant branch of research called *Internet of Things (IoT)* has grown up and meanwhile stucks into its teen shoes. According to the IoT vision, millions of devices such as sensors and actuators can communicate via Web-like structures through standardized software services. From a user and process perspective, these smart devices are resources that allow to measure or even change properties of entities of interest (i.e. a living room) in the real world. While the individual device used to communicate between the digital and the real world is interchangeable, rather the sensed (i.e. measure temperature) or even modified (i.e. activate cooling) *thing*[1] stands at the center of the application. Hitherto parallel, companies have been modeling their business processes from a process-oriented perspective for many years. Modern BPM systems automate these processes. They distinguish different phases in a life cycle. A fundamental phase before any process automation deals with process modeling. Its main goal is to create a model of the business process by applying a suitable language.

[1] The terms physical entity, entity of interest, object and thing are used replaceable.

Business processes that integrate the technologies of the IoT differ from conventional processes [2]. So far, modeling languages such as the industry standard BPMN 2.0, its compliant tools and its automating environments have offered only rudimentary support for expressing the component *thing*. With other words, the things of the Internet do not exist from the perspective of a BPM system. This is surprising, as the IoT promises to change not only our daily lives but also the business world significantly.

We suppose that conventional meta-models can be expanded by the missing concept *thing*. Its implementation shall empower end-users to model the things in business processes alongside to traditional concepts. This paper examines how the IoT domain component *thing* can be represented in the process model. For this purpose, we make the following contributions:

- Based on related contributions, we *identify three concepts* of the BPMN meta-model that are suitable for the representation of a thing when it comes to the execution of business process that involve IoT technology.
- Based on the IoT terminology and its domain model [1] we *define detailed requirements* for the new component.
- We investigate to what extent the identified BPMN concepts meet the defined requirements. *For each concept* we introduce a potential *BPMN extension* "Physical Entity" in order to meet all requirements that could not yet be covered by standard elements.
- By *evaluating the extension* we identify the extension "Custom Participant" as the most appropriate thing-representation.
- By further assessing the Custom Participant, we come up with a *solution beyond the BPMN standard*.

2 The Problem

The main components of the IoT are defined in a reference model [1] that potentially may perform tasks in business processes. The central component of these concepts is the *thing*, also named *physical entity*. To integrate the areas IoT and BPM seamlessly with one another, it should be possible to transmit all major components of the IoT meta-model to a corresponding meta-model of the BPM domain, including the concept *thing*. When examining different process modeling standards, it becomes clear that such a BPM counterpart does not exist. Likewise, it is not surprising that the *things* of the IoT are not part of the meta-model of the extensive industry standard BPMN either. This becomes a problem when it comes to the modelling and subsequent dynamic execution of elementary IoT-aware processes following the traditional BPM life cycle.

We consider the following process example: A Web service shall measure the temperature of the physical entity chocolate *by means of one currently available device that is accessible via the Internet.*

The product *chocolate* is available as a digital representation, but it remains unclear how it can be taken into account in a BPMN model. According to the IoT domain model [1], the problem area can be structured into four architectural components: the device, the thing, the native service and the IoT service. A *device* in the IoT is a technical artifact that can bridge the physical with the digital world. This connection is enabled via special on-device services (*native services*) such as sensing or actuating abilities. The device can communicate with other devices and is part of a physical construction unit. A *thing* is an identifiable, separable part of the physical environment which is of particular interest for a business process. Thus, a thing can become part of the digital world, if the artificial relation "attached to" is created between a device and the concerned thing (e.g. between the temperature sensor and the chocolate). *IoT services* are software components with standardized interfaces that expose the native interfaces of heterogeneous devices. They augment the functionality of one or more native services. By their well-defined interfaces, they denote an integratable part of a business process and can be bound to a process activity.

BPMN comes with a multitude of components of which some are potentially suitable for the constitutive thing integration. Anyhow, a detailed analysis and a comprehensive solution to the problem, both conceptually and as an implemented standard extension are still missing for the research community as well as for modelling users.

3 The Things in BPMN

The IoT comes with numerous of things being measured and influenced by devices that are able to flexibly perform as resources in business processes in a constantly changing environment. We aim at integrating this potential with traditional BPM systems which focus on executing planned processes with a constant set of resources. Existing BPM environments support a comprehensive lifecycle. One central and initial part of each lifecycle is the creation of a business process model. In order to bring the new IoT element *thing* to the envisioned BPM environment, we aim to provide a process model that includes the *thing* element, as a basis to express this new information. There are various Business Process Notations available, but [2] evaluated the industry standard BPMN 2.0 as the most IoT-aware state-of-the-art process modeling approach. The process model of BPMN comes already with a graphical and a machine-readable notation. The latter can comprise technical details [3] and is executable by a compliant engine. The process model is the outcome of the process design phase and serves as clearly defined interface between the design, resolution and execution phase. It shall cover typical and all needed constructs with the thing element.

4 Details of the Integration

The work of [4] provides a starting point for our work. It identifies three suitable concepts in the BPMN meta-model that can be used for such integration. These concepts are `Text Annotation`, `Data Object` and `Participant`. We compare these three potential elements and evaluate them. To do so, we adopt the IoT terminology and its domain model [1] as a definition and define detailed requirements for the new component *thing* in a first step. Next, we realize a potential extension of the meta-model for each of the three standards concepts. This allows us to examine in detail if the requirements are met. Subsequently, we evaluate how many of the requirements could be met and how many extensions were needed for this purpose. We receive a best standard solution. Finally, we improve this solution by hypothetical extensions that would require a slight change of the BPMN standard.

4.1 Requirements

Tab. 1 lists the identified requirements for the modeling element Physical Entity and its relations. We focus on two aspects: On the one hand the process modeler should be given the ability to express all important entity components graphically and on the other hand the output model should be machine-readable for the resolution and execution environment.

Tab. 1 Functional Formalization Requirements of Physical Entity Element

No.	Requirement	Rationale
E.1	There must be a way of representing the Physical Entity in the graphical process model as a separate element.	The Process Modeler needs standardized rules to express the participation of a Physical Entity in an own element.
E.2	There must be a way of representing the Physical Entity in the machine-readable model as separate element.	The process resolution environment [5] needs a schema to identify participating Physical Entities.
E.3	The selected or extended element for representing the Physical Entity must support contentwise its intention.	Realized extensions shall not contradict the semantics of any BPMN element. [11]
E.4	The Physical Entity element must not be target or source of a sequence flow.	The Physical Entity is a passive element and can neither directly execute tasks nor be instantiated by arriving tokens.
E.5	The Physical Entity element must not be target or source of a message flow.	The Physical Entity does neither directly receive nor send messages.
E.6	The Physical Entity element must not be target or source of a data association.	The Physical Entity does not contain data objects or data stores.
E.7	The Physical Entity element must support associations as connection type both as target and as source.	The Physical Entity can be bi-directionally associated with an IoT service. [1]
E.8	The Physical Entity element must not contain any responsibility assignments for any process element.	The Physical Entity is a passive object that cannot overtake any execution responsibility.
E.9	The Physical Entity element must not be assignable to a pool or lane.	The Physical Entity is able to have multiple relationships to further process participants. It does not exclusively belong to one single participant that is responsible for the entity.
E.10	The Physical Entity must take part in the process collaboration of the XML between further participants.	The Physical Entity collaborates with activities of further process participants.
E.11	The Physical Entity element must allow being a multi-instance element. [4]	One entity can be an augmentation of several entities to which the process shall apply to. [1]
E.12	The description model of the Physical Entity must be expressible in the form	The work of [6] and [7] foresees an entity description model.

	of entity properties.	
E.13	The association between a Physical Entity and an IoT service must be expressible in the graphical and machine-readable process model.	Following [1], the Physical Entity can be associated with an IoT service. The process resolution environment [5] needs the association in a machine-readable form.
E.14	The direction of association between a Physical Entity and an IoT service must be expressible.	[1] distinguishes between the associations "monitors" (gaining entity-related states) and "acts on" (changing entity-related states).
E.15	One Physical Entity element must support multiple associations to different process activities in the same process model. [4]	During a process flow even at the same time different IoT services can gain and change states of the same Physical Entity.
E16	The indirect association between a Physical Entity and an IoT Device must be expressible in the graphical and machine-readable model.	In order to enable the resolution of [5] based on [1] and to provide domain support to the modeler, the association between Device and Physical Entity has to be mapped to the process model.
E.17	The indirect association between a Physical Entity and a Native Service must be expressible.	To provide domain support to the modeler, the association between Native Service and Physical Entity has to be mapped to the process model.

4.2 Three potential extensions

Since the structure of the three potential extensions is complex, we will summarize the analysis of suitable representations in BPMN for the Physical Entity, considering the defined requirements and rationales. Firstly, the three most similar elements of the BPMN standard are identified and whether the Physical Entity requirements will cope with them is reviewed. Secondly, a potential Physical Entity class extension under or above the individual element class is discussed. In order to keep the BPMN extension conformance, standard classes are not changed.

Text Annotation and Establishment of Custom Artifact.
`TextAnnotation` is a subclass of `Artifact`. `Artifact`s are used to specify process-related information in the model, which does not affect the sequence or message flow. `TextAnnotations` define a mechanism to add additional descriptive information to the chart, but they are also represented in the machine-readable model. By a non-directional connection `TextAnnotations` can be connected to each object. Graphically they comply with data connections, but the machine-readable model differs in its output. One disadvantage of applying `TextAnnotations` is that the direction of the association is not modifiable and equates to type "none". In addition, it is impossible to forbid that a `TextAnnotation` is assignable to process flow elements contained by a Pool, since this is principally admitted for `Artifact` classes. A multi-instance property is not expected for Artifacts. As all existing elements, neither a `TextAnnotation` nor an `Artifact` depicts a separate `Physical Entity` element and they are not designed to tie-in to a description model. In parallel to the `TextAnnotation` class, BPMN provides an extension mechanism to

create own `Artifacts`, which resolves some of the shortcomings. This approach also has the disadvantage that a `Physical Entity` element shall not be assignable to further Pool and Lane containments that cannot be resolved by this class introduction. **Fig. 1** shows the respective class diagram.

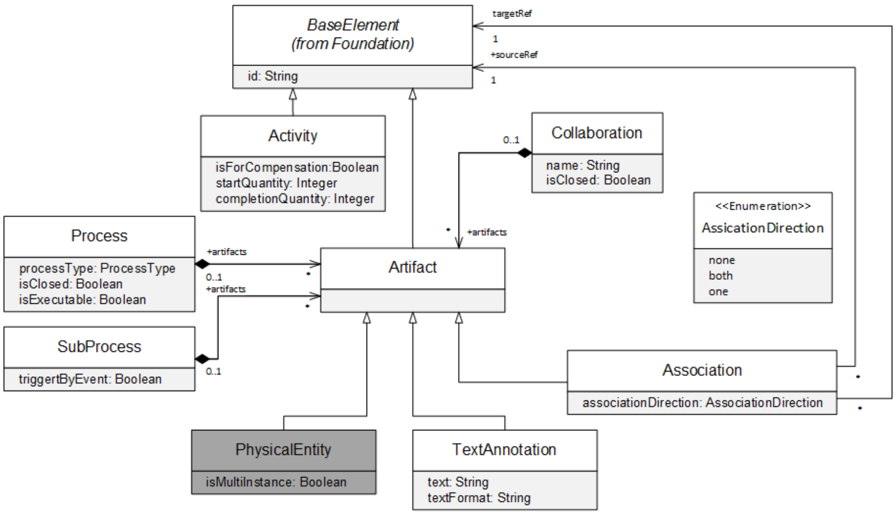

Fig. 1. Artifact subclass extension of BPMN standard

Data Object and Establishment of Custom Item Aware Element.
In the BPMN standard, `DataObject` is a subclass of `ItemAwareElement` that is applied to support the process execution. It is used to represent information flowing through the process. As a `FlowElement` the `DataObject` always belongs to a process or sub-process and, being an `ItemAwareElement` at the same time, it can reference a data item and a state definition. In a conventional manner, the class `ItemAwareElement` is devoted to detect data structures that are queried, transferred or changed during execution time. This contrasts with the task of a Physical Entity: in its passive role, it is not directly relevant to the final process execution, but rather, it is solely used for the resolution in the actual initialization of a process model. If the process is resolved as envisioned by [5], the Physical Entity is no longer needed. Nevertheless, the introduction of a new subclass of `ItemAwareElement` such as suggested by [4] is further examined (c.f. **Fig. 2**). Though `ItemAwareElement` initially does not support any sequence or message flows, it may support data connections if it is of the sub-class `DataObject`. In the XML output this inevitably leads to the fact that, once an `ItemAwareElement` contains an association it belongs to the class `DataAssociation`, and not to the class `Association`, because the specification of the graphical model contains the same symbols for both classes. To face this problem [4] suggests implementing the class `PhysicalAssociation`. This approach in turn leads to redundancies in the meta-model, as `PhysicalAssociation` and the standard class Association do not differ in their meaning. Additionally, a new composition-relation between `Collaboration` and

PhysicalEntity needs to be established to enable the allocation on the process participant level in the XML mode.

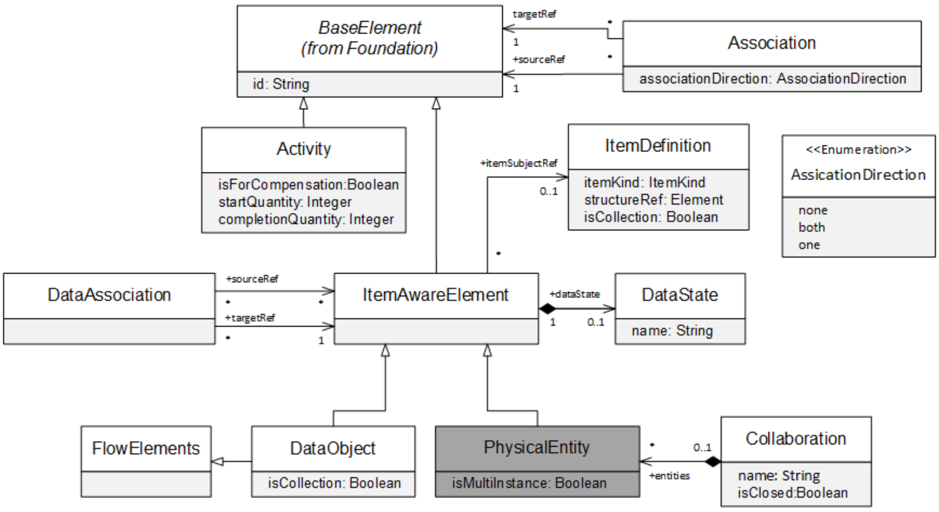

Fig. 2. Item Aware Element Extension of BPMN Standard

Participant and Establishment a Custom Participant.

Participant is a subclass of BaseElement of the BPMN standard and serves as a partner element in Collaboration - the representation of a process interaction with one or more Participants. Graphically, a Participant is represented as a Pool and takes over the task of a container for FlowElements. A special kind of a Pool is the Collapsed Pool containing no elements. Following [9], a Collapsed Pool is used to represent a black box pool: i.e. a Pool without any process reference. Consequently, a Collapsed Pool is either a pool in which FlowElements are unknown, or that simply does not have any FlowElements. The second option would be tantamount to a process participant who has no active execution responsibility, which is consistent with the properties of the PhysicalEntity. For the supplementary definition, the BPMN standard includes the PartnerRole (e.g. product) and the PartnerEntity (e.g. chocolate). With one of these two partner elements the Pool can be designated. A Participant already includes the option to be defined as a multi-instance element, to have associations, and to be neither a source nor target of a sequence flow or data association. However, the PhysicalEntity differs from other black-box process participants in the sense that it can never become part of a message flow. A PhysicalEntity is a passive participant whose state can be measured or changed by active process resources. Besides the process subscription, the entity does not take over any task or responsibility. By introducing PhysicalEntity as a subclass of Participant (cf. **Fig. 3**) a separate element is created, which meets the further criteria except that it is still possible to specify message flow connections for the PhysicalEntity.

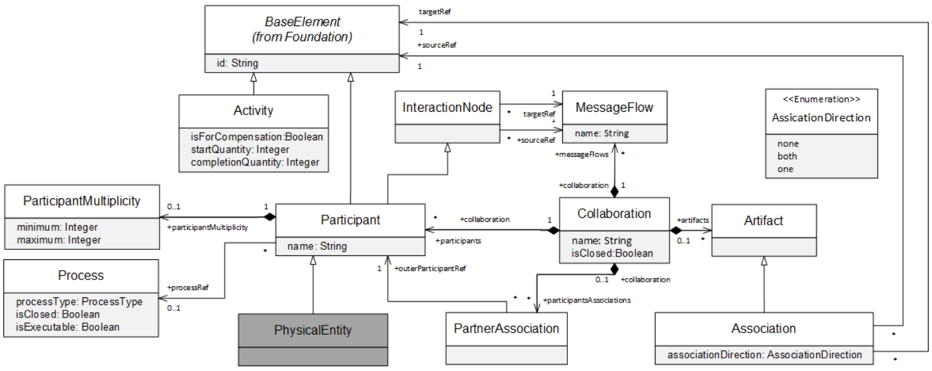

Fig. 3. Participant sub class extension of BPMN standard

4.3 Assessment

Tab. 2 summarizes whether the defined requirements for a separate `PhysicalEntity` element are satisfied for the three BPMN standard elements `TextAnnotation`, `DataObject` and `Participant`, as well as for their related extensions `CustomArtifact`, `CustomItemAwareElement` and `CustomParticipant`. The last two lines of the table provide the number of requirements that were met from all requirements and how many extensions were needed to obtain the best possible result. The legend is located below the table. Finally, the outcome is that the BPMN Collapsed Pool is the most appropriate standard element, fulfilling 13 out of the 16 points. In the case that no IoT-specific extensions are available, the Collapsed Pool should be picked for representing a Physical Entity in a Business Process. The following section presents a BPMN standard-compliant extension that introduces a meta-model sub-class below the class `Participant`. This extension even allows for meeting all requirements except the one excluding the definition of message flows.

Tab. 2: Entity requirement fulfillments of BPMN elements

No.	**Text Annotation / Custom Artifact**		**Data Object / Custom Item Aware Element**		**Participant / Custom Participant**	
E.1	-	+	-	+	-	+
E.2	-	+	-	+	-	+
E.3	+	+	-	-	+	+
E.4	+	+	+	+	+	+
E.5	+	+	+	+	-	-
E.6	+	+	-	-	+	+
E.7	+	+	+	-	+	+
E.8	+	+	+	-	+	+
E.9	-	-	-	+	+	+
E.10	+	+	-	O	+	+

E.11	-	O	+	O	+	+
E.12	-	O	-	O	-	O
E.13	+	+	+	+	+	+
E.14	-	+	+	+	+	+
E.15	+	+	+	+	+	+
E.16	+	+	+	+	+	+
E.17	+	+	+	+	+	+
Degree of fulfilment	11/17	16/17	10/17	13/17	14/17	16/17
Number of extensions		2		3		1
Not fulfilled: - Fulfilled, introduced: O Fulfilled, by default: +						

4.4 Solution Proposal

A solution for meeting all requirements can only be achieved through a more fundamental change in the BPMN meta-model. Based on the results of the previous assessment, we suggest a new element, `PhysicalEntity`, meeting all requirements. Therefore we present both a separate graphical element that is anchored on the BPMN Collapsed Pool concept, and a machine-readable element that enhances the introduced `Participant` subclass without being restricted by the BPMN extension conformance.

Graphical Model.
To integrate a separate `PhysicalEntity` element to the process model, the closest related BPMN standard element, `Collapsed Pool`, is extended graphically, representing a `Participant` without having any process reference in the semantic model. We suggest using the same significant empty pool shape that should be labeled with the corresponding name of the `PhysicalEntity` (e.g. chocolate). An icon within the `PhysicalEntity's` pool can be displayed before the lettering to identify the special process role of the `PhysicalEntity`. This approach is similar to the one used in the specification in order to describe the activity character with the aid of a meaningful marker. Based on the poster of [10], we propose using a small cow when selecting a self-explanatory marker, expressing that the `PhysicalEntity` represents a real-world entity that can even be alive. In comparison to the standard Collapsed Pool, the `PhysicalEntity` is not expandable, despite the way how it is realized by some tool implementations.

Fig. 4 shows a graphical process model containing two process participants in collaboration. As advocated by [9], we label the process pool with the process name “IoT Process”, and it contains process flow elements. The Collapsed Pool “chocolate” is of type `PhysicalEntity` and cannot be further extended since it is empty. The associations of the IoT-specific activities contain a direction that show the orientation of the association of the `PhysicalEntity's` state:

- From the Physical Entity to the Sensing Task: measuring of the entity state ("monitor")
- From the Actuation Task to the Physical Entity: setting the entity state ("act on")

While an Actuation Task with an associated Physical Entity acts as information sink, a Sensing Task with an associated Physical Entity acts as information source. Nevertheless, the actual flow of information originates/terminates not at the entity itself, but at the IoT Device, which justifies the type of connection between entity and activity as an association, rather than a flow of information.

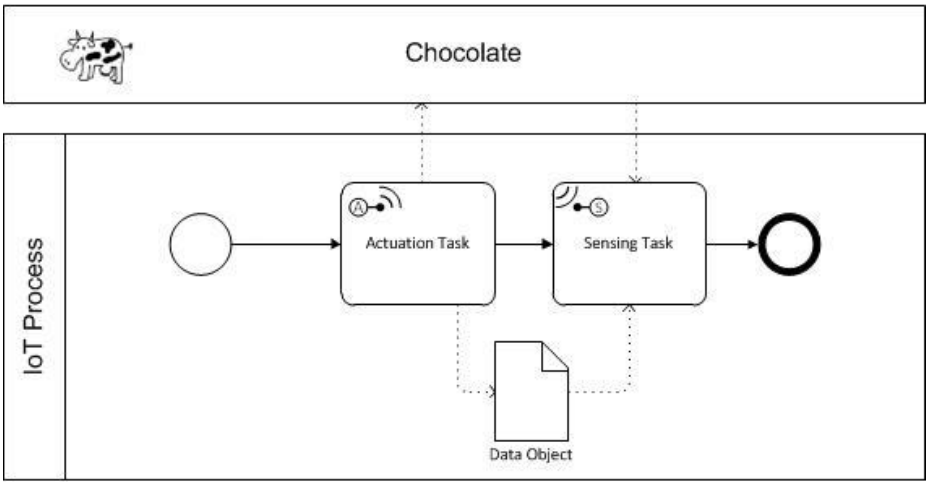

Fig. 4. Graphical process representation of Physical Entity

Machine-readable Model.

Based on the Participant extension discussion, in this subsection we come up with a solution beyond the BPMN standard to represent the Physical Entity as Participant in the CMOF meta-model. In contrast to the standard-conforming BPMN extension of the former subsection, this solution enables users to meet the remaining requirement of the message flow, so that a Participant of type Physical Entity cannot directly send or receive messages.

Fig. 5 shows the BPMN meta-model with the integration of the proposed extension. The shaded diagram areas represent those concepts that are newly added or changed, while the light areas belong to the unchanged BPMN meta-model. The new abstract class `ParticipantContainer` is added and derived from `BaseElement`. This class is used as a superclass for specific types of participants. `ParticipantContainer` contains the two subclasses `Participant` and `PhysicalEntity`. All attributes and associations of the old BPMN class `Participant` (c.f. **Fig. 3**) are attached to `ParticipantContainer`, except for the associations `processRef`, `interfaceRef` and `endPointRef`. These associations are not needed anymore, since the subclass `PhysicalEntity` never contains process elements and, therefore, none of the references. Given that the new class Participant (c.f. **Fig. 5**) inherits all properties of its superclass, and due to the multiple inheritances, it is a sub-class of `InteractionNode` at the same time, whereby all old properties remain unchanged. In the graphical model, a Participant can still be represented by both a Collapsed Pool and an Extended Pool, depending on whether it references a process. A Participant correspondingly still supports one or more message flows. In contrast to that, the class PhysicalEntity cannot contain any message flows. This is ensured by not deriving PhysicalEntity from InteractionNode. The newly added and changed standard attributes and associations refer to the definition for the BPMN 2.0 classes PhysicalEntity, Participant, Collaboration and ParticipantAssociation defined by [11].

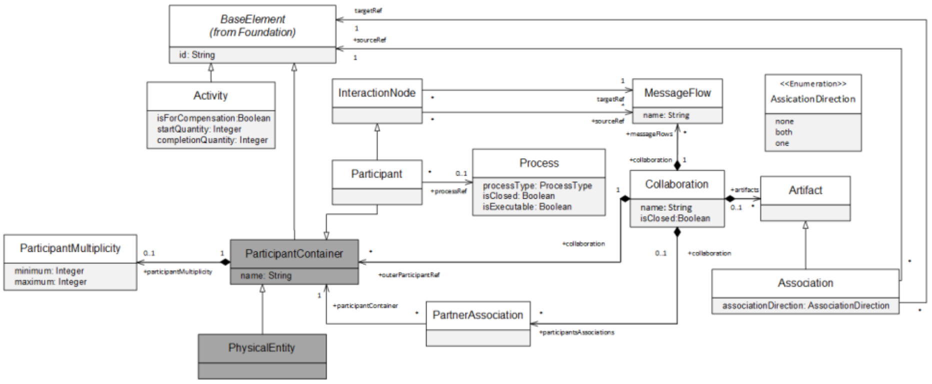

Fig. 5. Non-standard-conform BPMN extension of Physical Entity

Benefits.

The extension provides the possibility for the process modeler to uniquely represent the Physical Entity element in the graphical and machine-readable process model. The implementation was realized as close as possible to the standard without contradicting conditions and restrictions. The relations between the elements IoT Device, IoT Service and Native Service from [1] can still persist in the process model implementation. The character and intention of the Physical Entity element is kept by respecting the individual properties of the process, data and message flow. This extension proposal leaves open how the resolution of the process model is realized, but assumes that an automatic resolution approach such as envisioned by [5] is applied. This approach allows for the dynamic adaption to the changing availability of Physical Entities and attached IoT Devices. As some research efforts ([6], [7]) suggest semantic models for the description of some of the IoT-specific elements like the Physical Entity, we see the creation and integration of a IoT-specific model to the process modelling notation as a separate problem. Anyhow the problem is related to the refinement of the Physical Entities' representation.

5 Related Work

In this section we briefly summarize existing ideas of related research initiatives to express the IoT concept Physical Entity in the BPMN process model. Building on this groundwork, we identify problems: [4] states that the Text Annotation to an activity is the state of the art approach for expressing a Physical Entity in the process model. In order to cover the Physical Entity it is proposed to derive a new sub-class from the ItemAwareElement class, called PhysicalObject. In order to connect a

PhysicalObject to an activity, it is proposed to introduce a separate association type called PhysicalAssociation derived from the BaseElement class. This work was examined (above) to see whether a subclass of ItemAwareElement can meet the postulated requirements for a Physical Entity element. In comparison with [5], [4] separates the concepts IoT Device and Physical Entity more clearly. [5] uses the expression "entity" differently and does not clearly distinguish between the terms IoT Device and Physical Entity as it is envisioned by [1]. Nevertheless, a process example includes the Physical Entity "parcel" modeled as a collapsed pool, and representing a multi-instance process participant. In this case [5] doesn’t distinguish between the device itself in form of a tag and the Physical Entity parcel, but abstracts both concepts to the entity parcel. From this perspective, it seems reasonable that a parcel acquires the ability to communicate. This approach is, however, less reasonable for process models representing IoT Devices [12] as a maximum of one process resource existing in parallel to the Physical Entity element. Thus, the device and the entity are clearly separated concepts with their own semantic representation. The used devices, as well as the entities, are of central importance to the business process and cannot be considered in an augmented way.

6 Conclusion and Further Work

The absence of modeling concepts to directly express the *things* of the Internet as elements in a business process model is a significant obstacle to successfully resolve and automatically execute business processes of traditional BPM systems across distributed and Web-integrated devices. With this paper we have identified and investigated to what extent three different modeling elements of the industry standard BPMN are suitable to cover the specificities of the concept *thing*. In order to express the *thing* as an own element fulfilling all defined requirements; we introduced and evaluated for each of the concepts a standard-compliant BPMN extension. For each extension we presented the CMOF meta-model. As a result of the evaluation we conclude that a custom Participant of the semantic model visualized as a Collapsed Pool in the diagram is the closest standard-conform extension to express a thing in a business process.

Our future work will include investigating the significance of using IoT technology in business processes from a sustainability perspective. For this purpose, we will access the life cycle of an IoT-aware business process model towards the application of different wireless communication technologies (e.g. Wifi, Zigbee, Z-Wave, Bluetooth) building on the IoT Reference Architecture [1].

Acknowledgments. The authors would like to thankfully acknowledge the valuable support for this work provided by Alexandre de Spindler, Carsten Magerkurth and Jacques Pasquier-Rocha.

References

1. Carrez, F. et al.: Final architectural reference model for the IoT v3.0. EC FP7 IoT-A Deliverable 1.5 (2013)
2. Meyer, S., Sperner, K., Magerkurth, C., Pasquier, J.: Towards modeling real-world aware business processes. In: Proceedings of the Second International Workshop on Web of Things, p. 8. ACM (2011)
3. Freund, J., Rücker, B., Henninger, T.: Praxishandbuch BPMN. Hanser (2010)
4. Sperner, K., Meyer, S., Magerkurth, C.: Introducing entity-based concepts to business process modeling. Business Process Model and Notation, 166–171 (2011)
5. Concepts and Solutions for Entity-based Discovery of IoT Resources and Managing their Dynamic Associations. EC FP7 IoT-A Deliverable 4.3 (2012)
6. De, S., Barnaghi, P., Bauer, M., Meissner, S.: Service modelling for the Internet of Things. In: Computer Science and Information Systems (FedCSIS), Federated Conference on IEEE (2011)
7. De, S., Elsaleh, T., Barnaghi, P., Meissner, S.: An Internet of Things Platform for Real-World and Digital Objects. Scalable Computing: Practice and Experience 13(1) (2012)
8. Concepts and Solutions for Entity-based Discovery of IoT Resources and Managing their Dynamic Associations. EC FP7 IoT-A Deliverable 4.3 (2012)
9. Silver, B.: BPMN method and style. Cody-Cassidy Press (2009)
10. The things in the internet of things. Poster at the (IoT 2010). Tokyo, Japan, November (2010)
11. Business Process Model And Notation (BPMN). OMG Specification. Object Management Group (2011)
12. Meyer, S., Ruppen, A., Magerkurth, C.: Internet of Things-aware Process Modeling: Integrating IoT Devices as Business Process Resources. In Proceedings of 25th International Conference on Advanced Information Systems Engineering, Valencia, Spain (2013)