Patterns for IoT-based Business Process Improvements: Developing a Metamodel

Christoph Stoiber and Stefan Schönig
Professorship for IoT-based Information Systems, University of Regensburg, Germany

Keywords: Internet of Things, Business Process Improvement, Business Process Management, Patterns, Metamodel.

Abstract: The number of Internet of Things (IoT) devices is constantly growing across all areas of private and professional life. Especially industrial organizations are increasingly recognizing the IoT's disruptive capabilities and potential benefits for business processes along all value chain activities. In this regard, the integration of IoT technology into existing business processes enables valuable Business Process Improvements (BPI). However, it often remains unclear which BPIs can be expected by organizations and how the anticipated BPIs are realized in detail. Furthermore, the integration of IoT technology into existing business processes constitutes a major challenge caused by a lack of supporting methods, models, or guidelines. The paper at hand addresses this research gap by providing a metamodel that enables the illustration of generic IoT-based BPI patterns. It contains all relevant elements that are comprised by IoT applications with BPI propositions and can be used by industrial organizations as blueprints for conducting IoT projects. The metamodel development follows fundamental principles of design science research (DSR) and is extensively evaluated by deriving a first set of patterns from real-life IoT applications of three market-leading corporations. In addition, an expert survey is conducted to assess the metamodel's usefulness.

1 INTRODUCTION

Internet of Things (IoT) applications are omnipresent and influencing all facets of everyday life by providing disruptive technologies for private households and businesses of all kinds. Besides various smart home, smart grid, and smart city applications, especially industrial organizations can remarkably benefit from integrating IoT technologies into their business processes. The transformation of analog information into digital data, which can be processed worldwide in real-time, enables new business models, revolutionizes existing ones (Ng and Wakenshaw, 2017), and improves organizations' competitive advantage (Li, 2012). Moreover, the generation and use of comprehensive process data and the connection of process entities can be used to improve all types of business processes and thus optimize value creation (Del Giudice, 2016). Therefore, the integration of IoT technology into existing business processes can lead to beneficial Business Process Improvements (BPI) that are highly relevant for process-oriented organizations (Janiesch, 2020). For instance, equipping in-stock products with simple radio-frequency identification (RFID) tags can fundamentally enhance the traceability of warehouse processes and enable manifold further opportunities for improving downstream operations. Hence, the pressure on enterprises to integrate IoT technology into their processes is growing steadily, to the point that enterprises which don't adopt IoT, may not be competitive in the near future (Liu, 2017). However, a survey of more than 500 business executives revealed that 90% of industrial organizations are remaining in the proof of concept or even early-stage planning phases for IoT projects (Bosche, 2016). Knowing about the relevance of IoT technology integration, this seems rather surprising and indicates the existence of severe challenges for successfully integrating IoT technology into process landscapes. One main reason for this lack of IoT application maturity may be the complexity and heterogeneity of the used systems. Here, different technologies must be combined, e.g., various sensors and communication protocols, and integrated into the organization's existing information systems (Sethi and Sarangi, 2017). Another reason is the existing discrepancy between the organizations' expectations of IoT projects and the actual results (Skaržauskienė and Kalinauskas, 2015). Decision-makers need to have an explicit understanding of the value they can

Stoiber, C. and Schönig, S.
Patterns for IoT-based Business Process Improvements: Developing a Metamodel.
DOI: 10.5220/0011059300003179
In *Proceedings of the 24th International Conference on Enterprise Information Systems (ICEIS 2022) - Volume 1*, pages 655-666
ISBN: 978-989-758-569-2; ISSN: 2184-4992

expect and the technological aspects that are required to achieve it (Reijers and Liman Mansar, 2005). Finally, the "Act of Improvement", i.e., how existing business processes are transferred to the improved target state by integrating IoT, can often not be defined precisely. This fact reduces the plannability and thus the chance of a successful IoT project (Forster, 2006). To tackle these challenges, organizations need structured models that display and describe all relevant components of IoT-based BPIs. These models need to be generic enough to be applicable for similar scenarios and detailed enough to effectively guide organizations during the implementation of individual IoT-based BPI projects. In this regard, we define the term “IoT-based BPI” as the purposeful use of IoT technology within business processes to improve the same with respect to predefined objectives. Therefore, we formulate the following research question (RQ):

- **RQ1:** How can industrial organizations be supported at the identification and implementation of IoT-based BPI projects?

One auspicious approach to address this RQ is the development of generic patterns. Patterns are reusable artifacts which address a problem within a certain context by providing a suitable solution (Alexander, 1977). In this context, patterns can represent templates for IoT-based BPIs and are reusable for different kinds of industrial organizations (Forster, 2006). Using patterns can reduce the risk of IoT projects as well as support organizations with the identification of possible BPI potentials and the required IoT technologies, making them extraordinary valuable. Furthermore, all relevant application elements such as underlying problems and challenges, industry examples, performance indicators, or specific characteristics of the technical solution are provided. The prerequisite to formulate these patterns is an appropriate metamodel that displays basic design principles. The metamodel ensures completeness and consistency of the pattern descriptions and specifies their structure (Falk, 2013). Against this background, we formulate an additional supporting RQ:

- **RQ2:** Which metamodel can enable the illustration of generic yet adoptable IoT-based BPI patterns?

The paper at hand addresses both RQs by proposing a metamodel which can be used to create IoT-based BPI patterns. The metamodel design is based on the design science research (DSR) methodology by Peffers et al. (2007) including an evaluation according to the Framework for Evaluation in DSR (FEDS) of Venable et al. (2016).

The remainder of this paper is structured as follows. Section 2 presents the theoretical foundations of the disciplines IoT and BPI as well as an overview over the concept of patterns and metamodels in information systems research. In section 3, the underlying research methodology is described that has been applied for developing and evaluating the metamodel. Subsequent, the design and development phases are illustrated in section 4. Section 5 presents the summative evaluation of the metamodel, concluding with a summary, discussion, and the description of limitations in section 6.

2 THEORETICAL FOUNDATION

2.1 Internet of Things Meets Business Process Improvement

There are dozens of different approaches for defining IoT, its components, features and capabilities, and the *things* itself. The Institute of Electrical and Electronics Engineers (IEEE) combined several different descriptions, explanations, and characterizations towards a universal definition. According to the IEEE, IoT is a network that connects uniquely identifiable things to the internet. Through the exploitation of unique identification and sensing, information about the *thing* can be collected and the state can be changed from anywhere, anytime, by anything (Minerva, 2015). The term *thing* therefore corresponds to the idea of creating a ubiquitous presence of objects which are equipped with sensors, actuators, or tags. On the other side, the term *internet* refers to the ability of these things to build a network of interconnected objects based on several specific network technologies. These two perspectives can be complemented by a semantic view, which represents the ability of IoT to uniquely identify things and store, process, and exchange data (Atzori, 2010). Current research and already implemented applications now show that IoT technology reveals many extensive possibilities for improving business processes (Stoiber and Schönig, 2021). In this regard, especially redesigning and therefore improving business processes is a timely and relevant topic in both research and business environment and is considered as one of “the most important and common titles in both literature and applications” (Coskun, 2008). Despite IoT’s capabilities to enhance BPI and therefore sustainably optimize the organization’s overall performance, there is a lack of research regarding IoT-based BPI. Among the limited number of contributions, Janiesch al. (2020) created an overview of existing research and remaining challenges. Here, especially the need for further research on how to benefit from the integration of IoT into business processes has been highlighted. This research gap can be tackled by developing a metamodel that enables the creation of patterns and adds to the descriptive knowledge of IoT-based BPI. This approach has been proven in several other research disciplines and is well received at organizations of all industry sectors (Winter, 2009).

2.2 Metamodels and Patterns in Information Systems Research

Patterns, initially described by Alexander (1977), describe a recurring problem or challenge in the real world and the basic features of the solution to this problem. This solution is generic enough to be applied to many similar problems without ever being implemented in exactly the same way. Although Alexander (1977) created this definition in the context of architecture, the idea of patterns is transferable to other domains, especially information systems research (Gamma, 1994). In the context of enterprise and systems modeling, Fowler (1996) described patterns as an idea that has been useful in one practical application and is likely to be useful in others. According to Gamma et al. (1994), patterns consist of four essential elements. First, the pattern must have a name for identification. Then there is a description of the problem, i.e., in what context the pattern might be useful. The third element is a description of the problem solution. This must not be done by a concrete solution, because the pattern should be applicable to different scenarios, but by a description of the interaction of different mechanisms that lead to a problem solution. Finally, the consequences of the pattern must be described, i.e., the positive and negative effects that can result from the application of the pattern. Depending on the purpose of the pattern, this basic description can be extended by further elements. There has been considerable research on patterns in information systems for more than two decades leading to several relevant approaches indispensable from a research and practical perspective. Beyond doubt, software development is one of the disciplines that benefited most from the creation of patterns (Winter, 2009). Here, patterns can support the design of individual object-oriented software components or assist with the composition of software components to applications (Schmidt, 2000). As this discipline includes complex tasks, patterns can bridge the gap between high-level integration plans and the actual implementation challenges by providing guidelines to compensate the lack of experience at decision makers (Hohpe, 2003). This leads to reduced time consumption and cost while improving the quality of project execution. Moreover, patterns can be used for process-related disciplines such as Workflow Management or Business Process Modeling (Kühn, 2005). For the discipline of BPI, the creation of specific patterns has barely been addressed in research. Reijers and Liman Mansar (2005) described a set of textual Business Process Redesign (BPR) best practices including a framework to classify them. Forster et al. (2006) built up a framework and toolset for creating and structuring BPI patterns while creating a first set of patterns. Another relevant contribution by Falk et al. (2013) proposes a metamodel that facilitates the illustration of BPI patterns. In this respect, patterns constitute models that are derived from an origin metamodel.

In general, a model can not only describe objects that exist in the real world, but also abstract constructs. If the abstract construct described is a model, the describing model is called metamodel (Gonzalez-Perez, 2008). The relationship between model and metamodel can also be referred to as a class-instance relationship. This is an analogy to object-oriented programming, where a class describes the attributes and methods of the objects to be formed from it, without itself being an object. By instantiation, objects or instances can be formed from the class, which in turn are mappings of real objects. A metamodel describes the types of model building blocks available, the types of relationships between the model building blocks, the rules for linking between model building blocks by relationships, and the semantics of the model building blocks and relationships (Ferstl, 2013). To create a metamodel, a suitable modelling language is necessary to represent and communicate relevant information about a model. Modeling languages are defined by their syntax, notation, and semantics. The syntax describes the elements of a modeling language and how they may be linked together, i.e., it describes the grammatical rules. The notation describes the symbols and characters that may be used to capture a model. Ultimately, the semantics determines how certain information is to be interpreted, e.g., when ambiguities occur in the model (Kühn, 2005).

2.3 Related Work

As described in subsection 2.2, there has already been research conducted on general BPI patterns and metamodels that do not particularly focus on IoT but consider BPIs of any kind. Especially noteworthy is the contribution of Falk et al. (2013), who created an explicit metamodel that enables the creation and formulation of BPI patterns and can be used as a template and basis for further research.

Moreover, the concept of patterns has also been applied to several topics related to IoT. As IoT technology consists of different layers, comprising perceiving, networking, or data processing technologies, a great variety of different patterns can be formulated that support system engineers with integrating whole applications into business environments. The design and architecture of IoT systems can eminently benefit from patterns that assist in designing scalable and replicable IoT applications (Washizaki, 2020). Another focus within this research area is on data exchange and network technology patterns along multiple connected devices, machines, or process entities (Reinfurt, 2016). However, the formulation of a metamodel for IoT-based BPI patterns has not been addressed yet.

3 RESEARCH METHODOLOGY

To tackle this research gap and answer the formulated RQs, we developed a metamodel that can be used to create and illustrate reusable IoT-based BPI patterns. To develop the metamodel as a DSR artifact, we followed the process model of Peffers et al. (2007). This proven method is based on the methodology of Hevner et al. (2004) and provides detailed phases to carry out DSR. It consists of six iterative phases in a nominal sequence including *i)* the identification and motivation of the underlying problem, *ii)* the definition of objectives of the solution, *iii)* the actual design and development, *iv)* the demonstration, *v)* an evaluation, and *vi)* the communication to an appropriate audience.

Initially, every conduction of DSR is based on a research entry point that necessitates and justifies the artifact development. For the paper at hand, the existing problems and challenges that organizations face at integrating IoT into their business processes constitute a problem-centered research entry point. Moreover, the lack of artifacts that support the realization of IoT-based BPIs necessitate the creation of a suitable DSR artifact. This research endeavor is of special interest, as the integration and use of IoT technology is an enabler for economic success and becomes increasingly important. The objective of the developed artifact is to provide a basis for the creation of reusable patterns of IoT-based BPIs which serve as blueprints and templates for organizations.

In contrast to creating a complete new metamodel from the scratch, the improvement and revision of an existing and thematically related metamodel enables the adoption of proven concepts and ideas. Therefore, the metamodel for BPI patterns according to Falk et al. (2013) served as the basis for development. It is generic enough to represent all patterns of IoT-based BPIs since these represent a subset of BPI patterns. However, it is not specific enough to appropriately illuminate the aspects of the IoT domain due to its complexity and unique features. For this reason, the base metamodel needed to be adapted with respect to IoT. Like in the original metamodel, a class diagram is used for modelling as it provides sufficient semantic expressiveness for metamodeling. To adapt the base metamodel, we performed two development iterations comprising methods of Grounded Theory and a Delphi study. Figure 1 shows both iterations, including data sources, the applied research methods, and the resulting metamodel classes after each iteration. To evaluate the final metamodel, we followed the framework of Venable et al. (2016).

Within the first design iteration, an explorative inductive approach has been selected. Hereof, an extensive systematic literature review (SLR) was conducted to investigate literature describing IoT applications with BPI reference. Subsequent, the found literature was analyzed following the Grounded Theory and its methods of open and axial coding (Corbin and Strauss, 1990). This enabled the identification of indispensable aspect of IoT-based BPIs which could be used to adapt the metamodel. Within the author team, we applied the method of inductive reasoning (Hempel, 1966) to critically discuss the findings and select the most appropriate metamodel adaptions. Within the second iteration, we included additional expert knowledge into the research approach. Hereof, we conducted a Delphi study with nine experts from industry and academia to consequently refine the metamodel. In four rounds, the experts were asked to rate and eventually adapt the metamodel based on their expertise of the research area. Gradually, the metamodel has been adapted by *i)* removing redundant elements, *ii)* adding additional required elements, and *iii)* retaining or slightly adjusting the remaining elements. Having refined the metamodel, we performed a summative evaluation to assess, if it adequately addresses and solves the formulated RQs. In this regard, we introduced the metamodel to the Linde plc and two other multinational industrial corporations. Seven practitioners from different departments were

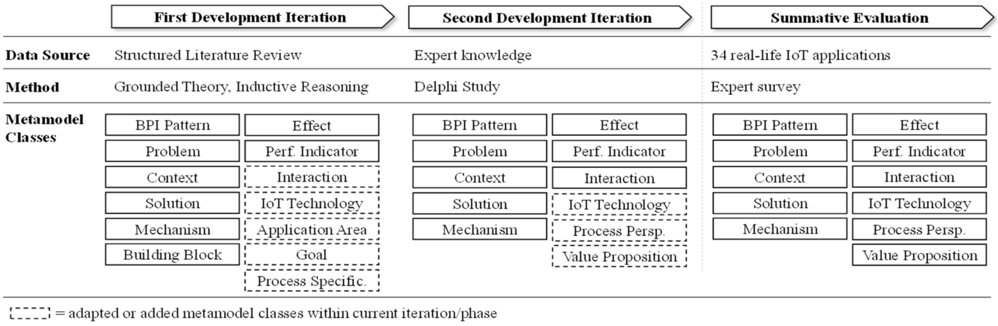


Figure 1: Development procedure.

asked to analyze a set of 34 IoT applications of their different business areas to derive patterns.

Subsequently, we conducted an expert survey on the practitioners to collect evidence and feedback. We used the results of the survey to assess the predefined evaluation criteria of *usefulness*, *conciseness*, and *robustness*. In the following section, the initial base model and all metamodel development iterations are described in detail.

4 METAMODEL DEVELOPMENT

4.1 Baseline Metamodel

The metamodel for BPI patterns of Falk et al. (2013) is illustrated as a class diagram, whereby each element of a pattern is represented by a specific class. The properties of these classes are described by attributes (Fowler, 1997), while relationships between the classes are represented by undirected binary associations and their multiplicity. This multiplicity specifies the relationships between the individual object classes. The central class of the metamodel is *BPI Pattern*, which is instantiated by a unique *Name* and an *Example* (cf. Figure 2). The name describes the overall purpose of the pattern and can be uniquely identified. In addition, there is the class *Problem*, which is defined by the attributes *Name*, *Description*, and the actual *Consequences* of the problem for the process.

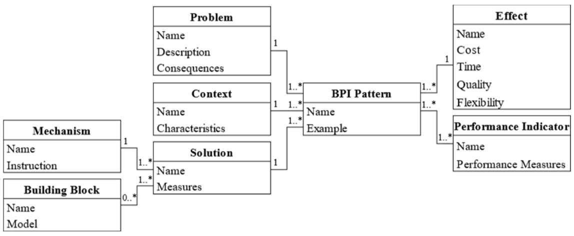


Figure 2: Base metamodel.

Each pattern addresses exactly one problem, but a specific problem can also be solved by different patterns. Furthermore, the *Context* class is directly related to *BPI Pattern*. It is explained by a *Name* and context-specific *Characteristics* and describes the required circumstances for the pattern to be applicable. As with the problem, each pattern exists in exactly one context, but multiple patterns can exist in the same context. Each pattern also contains a *Solution*, which is described by a *Name* and the *Measures* required to achieve the goal. The same solution can again be applied to multiple patterns, but each pattern has only one solution. Bound to the solution are one or more *Mechanisms*, each defined by a *Name* and precise action *Instructions*. In addition, a solution can optionally contain one or more *Building Blocks*. These building blocks are predefined models that can be implemented to solve the problem without customization. In addition, the pattern is related to an *Effect*, which is defined by a *Name* and the BPI dimensions *Cost*, *Time*, *Quality*, and *Flexibility* (Dumas, 2018). Finally, each pattern is related to one or more *Performance Indicators*. These are defined by a *Name* and *Performance Measures* that can be used to represent the improvement after the pattern has been implemented.

4.2 First Development Iteration

To adapt the base metamodel, we first performed an inductive development iteration. We decided to start with this approach, as a large number of IoT applications is available in scientific literature which can be used to identify additional metamodel classes. For inductive approaches, the information processing is performed from subsystems to form a perception of a top-level system. This aggregation of information is suitable to analyze initially unknown data relationships and transfer them to a generic metamodel. We followed the recommendations of Templier and Paré (2018) to identify a set of appropriate literature and subsequently extract relevant data. For the identification of literature, we performed a SLR according to the method of vom Brocke et al. (2009). To allow a rigorous search and improve the traceability of the literature selection process, the Preferred Items for SLRs and Meta-Analysis (PRISMA) statement has been applied. Initially, the search string *("IoT" OR "CPS") AND ("BPI" OR "Process Improvement" OR "Process Optimi?ation" OR "Process Automation" OR "Application" OR "Process Improvement")* and the written-out forms have been formulated. Figure 3 illustrates the results of the SLR as a PRISMA flow diagram. To incorporate and consider preferably all relevant journals and conference proceedings of the research area, ACM Direct Library, AISeL, IEEE Xplore, ScienceDirect, Scopus, and Springer Link have been queried. According to the PRISMA statement, four criteria were defined that a paper needs to achieve to be eligible for the SLR. The publication must *i)* be a peer-reviewed research paper published in a journal or conference proceeding, *ii)* propose an evaluated solution or real industry application, *iii)* have clear links to BPI, and *iv)* be relevant and up to date. As criteria *ii)* and *iii)* are assessed in a rather qualitative manner, criterion *iv)* is defined as a publication date after 2015 and a minimum number of 30 citations. The literature search and the included reference follow up resulted in the selection of 81 eligible publications.

Having identified the eligible sample of publications, we analyzed it and extracted relevant data using the Grounded Theory. In this regard, we applied the methods of open and axial coding, as proposed by Corbin and Strauss (1990). This approach enabled the derivation of metamodel classes and attributes from the sample of IoT applications.

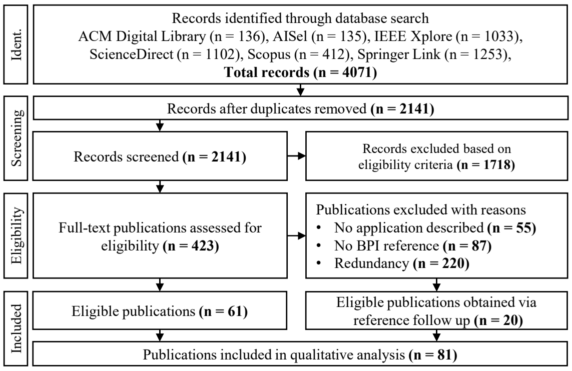


Figure 3: PRISMA flow diagram.

In the first round, each author analyzed 40 publications of the sample using open coding as an interpretive method to analytically break down all IoT-based BPI applications. The goal was to develop substantiate categories that enable a description, naming, and classifying. After this first round, we discussed the identified categories and harmonized the individual understanding of the main elements of IoT-based BPIs. In the second round, we applied the method of axial coding to relate the formulated codes to each other. This enabled the creation of further categories and subcategories. In a second discussion, the results were again harmonized. In round 3, the remaining 41 publications were coded with the created set of categories and subcategories to test them against data. Subsequent, we clarified and resolved any remaining coding differences. Following inductive reasoning according to Hempel (1966) we extensively discussed the created categories and subcategories to select the most potent and relevant ones for the metamodel adaption. These have been used to create a set of classes and related attributes which were added to the base metamodel.

4.3 Second Development Iteration

To refine the initial metamodel draft we performed a structured four-round Delphi study. A Delphi study is an iterative method to solicit information about a specific topic through the completion of several surveys (Loo, 2002). It has been widely used to combine expert knowledge and find group consent for complex issues that lack empirical evidence (Loo, 2002). For this reason, Delphi studies are highly present in the field of DSR research. The study process included the selection of experts with different backgrounds to minimize bias. They did not get introduced to each other, which led to more creative outcomes and reduced conflicts within the group as well as group pressure. The experts were asked to rate or validate the metamodel classes and attributes of the first draft. After each round, the results of all experts were consolidated and used for refinement. We formed a panel of nine experts including five practitioners and four researchers with expertise in the fields of IoT and BPM. The selected experts have working experiences ranging from four to 21 years. All experts have at least a bachelor's degree and are based in Germany, the US, or the Netherlands. Figure 4 shows the applied four-round Delphi study including all information flows between the authors, or facilitator, and the expert panel.

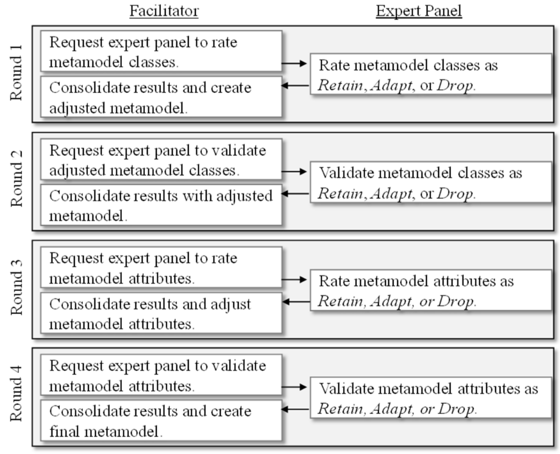


Figure 4: Delphi study design.

In **Round 1**, the expert panel was asked to rate the metamodel classes of the initial metamodel draft. They could *Retain*, *Adapt* or *Drop* the individual classes as well as *Add* further ones. The results of round 1 were analyzed and consolidated using a systematic decision tree which has already been used in different Delphi studies and proved to be appropriate (Serral, 2020). A class was only dropped, if more than 60% of the experts agreed on this option. No adaptions were considered, if the percentage to retain was at least 80%, while minor adaptions were performed for a retain rate between 60% and 80%. Major adaptions were needed if the retain rate was below 40% or at least 50% of the experts agreed on the option to adapt a class. In **Round 2**, the experts validated the results of the first round, followed by another consolidation phase. In **Round 3**, the expert panel was requested to rate the attributes of each class. For new classes, they were asked to introduce corresponding attributes. The consolidated results were validated in **Round 4**. After this round, a discussion with all experts helped to get feedback and gain insight into the background of the individual decisions. Having refined the classes and attributes, we analyzed relations and subsequently added multiplicities for all classes.

4.4 The IoT-based BPI Metamodel

The final metamodel for IoT-based BPI patterns consists of 11 classes and 28 attributes. During the first development iteration, we added five classes, namely *IoT Technology*, *Application Area*, *Interaction*, *Goal*, and *Process Specification*. During the refinement, two further classes *Process Perspective* and *Value Proposition* could be created. The previously added classes *Goal* and *Process Specification*, on the other hand, were removed as a result of the Delphi study. In addition, the class *Building Block* of the base metamodel was removed. Figure 5 shows the resulting metamodel including all classes, attributes, and relations which will be subsequently explained in detail.

According to Falk et al. (2013), the class *Building Block* can be used for result-oriented patterns, i.e., patterns that directly describe the target process, and are models that can be implemented without adjustments. In contrast, procedure-oriented patterns only describe instructions on how to improve the process, but no direct implementation. Since IoT systems are very complex and cannot provide any benefit without appropriate integration in the process, it is assumed that patterns for process improvement through IoT can only be procedure-oriented. Therefore, the expert panel agreed to delete this class from the metamodel. Also, we changed the multiplicity of the class *Mechanism*. In the base metamodel, each solution contained exactly one mechanism. However, this is an unnecessary restriction that makes it difficult for modelers to create domain-specific BPI models. By removing the restriction, it is possible to define further implementation details of the IoT system, while the modeler is given greater freedom. The first new class of the extended metamodel is *Interaction*. As part of the solution, it describes the *Human Involvement* in the IoT system. This is an essential aspect for describing the integration of the IoT system into the process and has already been discussed by Patterson et al. (2017). For example, it can describe whether a dashboard is only available to the process owner or whether every actor in the process is always provided with information via wearables, smartphones, or other devices. It comprises interfaces between the IoT application and humans regarding data input and output. Being a domain-specific element, it constrains the generic BPI metamodel to an IoT-based BPI metamodel. In particular, the information output or the information transfer to human actors had to be modelled previously using the class *Mechanism* or could not be modelled at all. Each solution can contain one or more *Interactions* as there might be several interfaces regarding data input or output, or different groups of persons might be affected. However, the class is not mandatory, as highly automated IoT systems might not have any human involvement at all. As another new class, *IoT Technology* has been added to the metamodel.

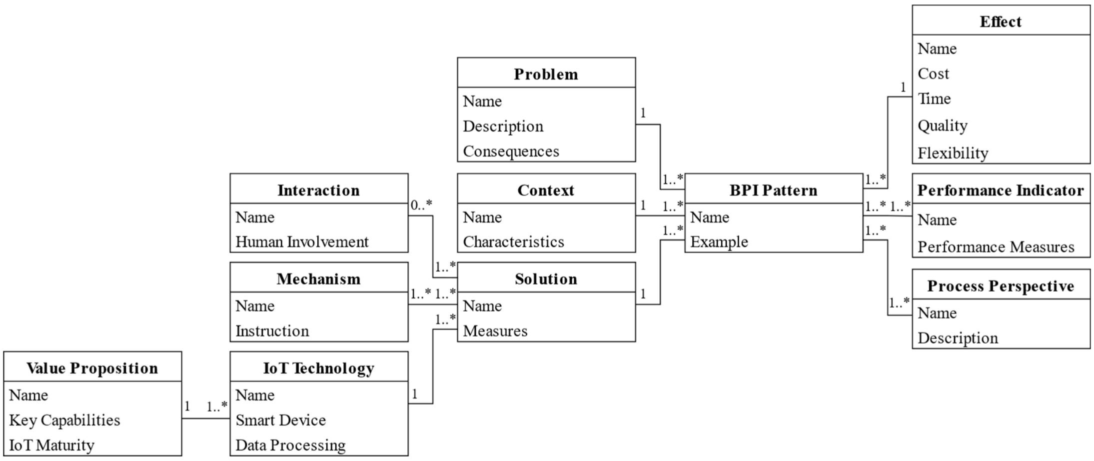


Figure 5: Metamodel for IoT-based BPI patterns.

As the base metamodel could not be used to represent these requirements, the aspects for the basic description of the technological requirements were combined under this generalized class. The class comprises two attributes that explain the necessary technological specification of the IoT application. At first, the *Smart Device* type reflects technological and architectural principles. As already described in subsection 2.1, sensors, actuators, and network technologies can turn conventional objects into smart things (or devices). These aspects can be displayed within this attribute. Kortuem et al. (2010), e.g., have already defined three different types of smart devices. Activity-aware devices understand events and activities causally related to the use of the object. Policy-aware devices can reflect whether activities and events are compliant with organizational policies and process-aware devices can place activities and events in the context of processes. A more detailed description of the required hardware, e.g., sensors and actuators, and networking technologies would be too concrete for the creation of generic patterns. The second attribute *Data Processing* describes the basic features of how the collected IoT data is analyzed and eventually used to improve the underlying business processes. With cloud computing, for instance, the IoT device is only responsible for generating the data and does not provide any data processing capabilities. In contrast to centralized data processing, edge computing involves processing and analyzing the generated data (or at least parts of it) directly at the edge of the network by specially designed devices. Depending on the application and the structure of the IoT system, hybrid approaches can be possible, too. Directly connected to *IoT Technology* is the new class *Value Proposition*. It describes the actual value that the IoT technology can provide to solve the addressed problem. It goes beyond the simple description of technological specifications but rather outlines, which disruptive features and capabilities the combination of sensors, networking, and data processing technologies enables. The first attribute that details the class is *Key Capabilities*. The IoT comprises novel and disruptive capabilities that distinguish it from other technologies. To enable beneficial BPIs, these capabilities must be profitably and systematically exploited. While the combination of these capabilities is often relevant for IoT-based BPIs, in most cases individual key capabilities can be identified that are particularly relevant. Examples for such capabilities would be universal scalability, comprehensive perception, embedded intelligence, or interoperability. By using specific IoT technologies and therefore exploiting a set of capabilities, the *IoT Maturity* can be defined. Maturity in this case refers to the complexity of an IoT application, how deeply it is embedded into the process, and how value is generated. It ranges from simple data collection and analytics to completely automated tasks within the process. Tai Angus Lai et al. (2018) have addressed this topic and identified different possibilities to define this IoT maturity. They stated situational awareness, decision-making support, information exchange, and autonomous systems as potential manifestations. Finally, the class *Process Perspective* was added to the metamodel. It describes the perspectives and therefore process aspects that are influenced most by the IoT application. This is especially useful to illustrate, how the IoT application affects and redesigns the process. Jablonski and Bussler (1996) have stated six process perspectives that can be used in this regard. The behavioral perspective comprises elements of the right process workflow or sequence, legal regulations such as reporting obligations, and internal requirements. The organizational perspective focuses on the personnel that is involved in the process execution. Its main components are responsible process owners, admins, and users. In addition, the underlying system is part of this perspective and represents for example the IT environment. The functional perspective includes the concrete process steps, tasks, and events. Most of the processes, especially in the industry, comprise several machines, tools, and software applications which can be described as the operational perspective. The data perspective involves all data and documents that are necessary for process execution. Finally, the locational perspective describes the specific locations of process entities, e.g., machines or workers.

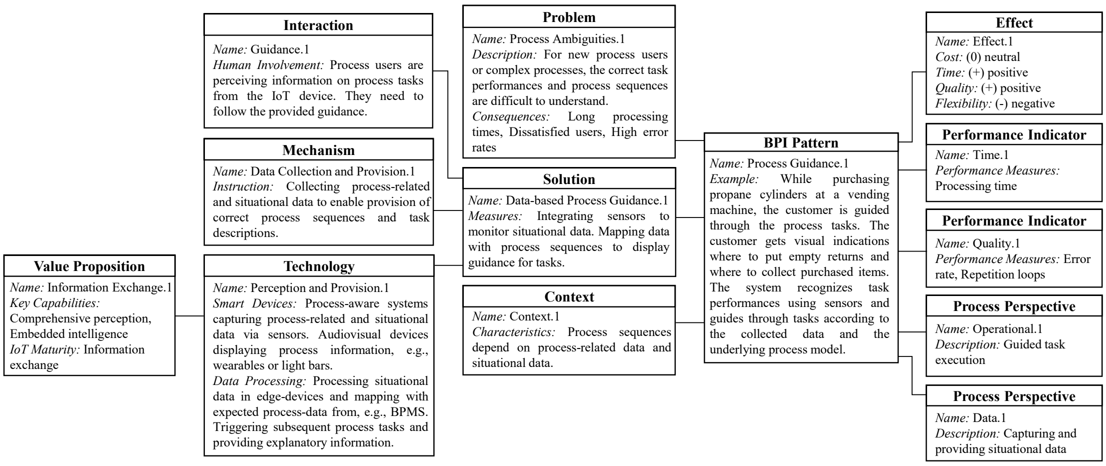
5 EVALUATION

5.1 Evaluation Setup

To be applicable for further research or industrial use, we applied the FEDS of Venable et al. (2016) to formulate an evaluation goal, derive evaluation criteria, and apply an appropriate evaluation method. To assess the goal achievement, we chose the evaluation criteria *usefulness*, *conciseness*, and *robustness*, as described by Prat et al. (2015). First, the metamodel must enable an appropriate derivation and description of patterns. Hence, it must comprise all classes and attributes that are required to illustrate generic abstractions of IoT-based BPIs. We translated this into the criterion *usefulness*. Furthermore, it must be appropriately detailed to depict different patterns, described as *robustness*. However, it also must be *concise* and generic enough to be applicable for a wide range of possible applications. The evaluation is performed by deriving a set of IoT-based BPI patterns and investigating if the created metamodel can adequately illustrate them. For deriving the patterns, we requested seven practitioners from the Linde plc and two other corporations to analyze IoT applications within their business areas. All practitioners have a broad knowledge in IoT technology and business processes in general and have working experiences of five to 21 years. They work as technical project managers, IT managers, automation experts, or digitalization managers while each of them has implemented at least two major IoT applications in the primary value chain activities of their corporations. In total, they identified 34 applications that were suitable for further analysis. In a joint workshop, six different patterns could be derived and illustrated using the provided metamodel. These patterns are *Process Guidance*, *Derivation Detection*, *Authentication & Authorization*, *Task Distribution*, *Proactive Activity Execution*, and *Activity Automation*.

5.2 Pattern: Process Guidance

The first pattern *Process Guidance* (see Figure 6) generically describes applications focusing on improved user guidance. By capturing situational and process-related data, the actual process state and subsequent process sequences can be ascertained. The next process tasks can then be displayed to process participants, e.g., via wearables. This pattern mainly affects the operational and data perspectives, as the way of performing the process tasks is changed by using input and output data. The used smart devices are process-aware as they need to capture process-related data, process it, and provide it to the process participants with respect to the current process state.



Value Proposition
Name: Information Exchange.1
Key Capabilities: Comprehensive perception, Embedded intelligence
IoT Maturity: Information exchange

Interaction
Name: Guidance.1
Human Involvement: Process users are perceiving information on process tasks from the IoT device. They need to follow the provided guidance.

Mechanism
Name: Data Collection and Provision.1
Instruction: Collecting process-related and situational data to enable provision of correct process sequences and task descriptions.

Technology
Name: Perception and Provision.1
Smart Devices: Process-aware systems capturing process-related and situational data via sensors. Audiovisual devices displaying process information, e.g., wearables or light bars.
Data Processing: Processing situational data in edge-devices and mapping with expected process-data from, e.g., BPMS. Triggering subsequent process tasks and providing explanatory information.

Problem
Name: Process Ambiguities.1
Description: For new process users or complex processes, the correct task performances and process sequences are difficult to understand.
Consequences: Long processing times, Dissatisfied users, High error rates

Solution
Name: Data-based Process Guidance.1
Measures: Integrating sensors to monitor situational data. Mapping data with process sequences to display guidance for tasks.

Context
Name: Context.1
Characteristics: Process sequences depend on process-related data and situational data.

BPI Pattern
Name: Process Guidance.1
Example: While purchasing propane cylinders at a vending machine, the customer is guided through the process tasks. The customer gets visual indications where to put empty returns and where to collect purchased items. The system recognizes task performances using sensors and guides through tasks according to the collected data and the underlying process model.

Effect
Name: Effect.1
Cost: (0) neutral
Time: (+) positive
Quality: (+) positive
Flexibility: (-) negative

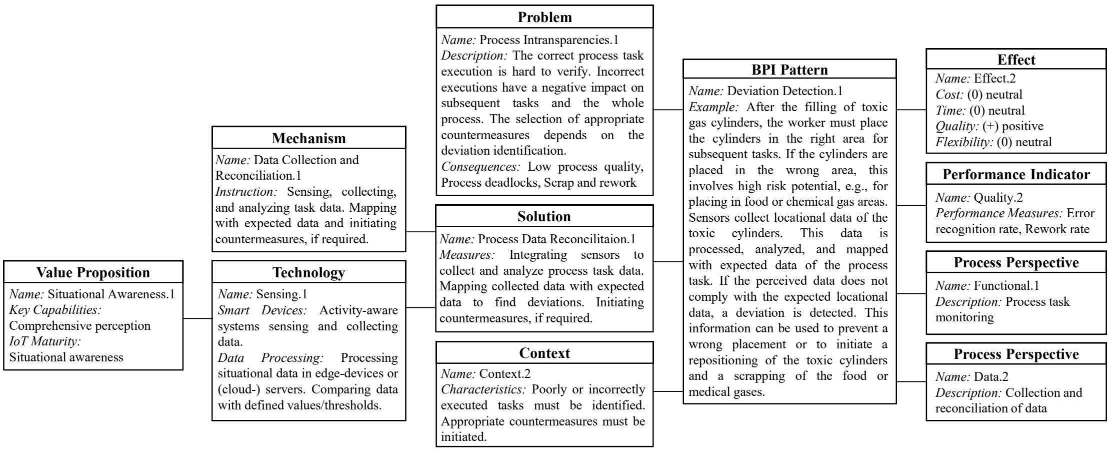
Performance Indicator
Name: Time.1
Performance Measures: Processing time

Performance Indicator
Name: Quality.1
Performance Measures: Error rate, Repetition loops

Process Perspective
Name: Operational.1
Description: Guided task execution

Process Perspective
Name: Data.1
Description: Capturing and providing situational data

Figure 6: Process Guidance pattern.



Problem

Name: Process Intransparencies.1
Description: The correct process task execution is hard to verify. Incorrect executions have a negative impact on subsequent tasks and the whole process. The selection of appropriate countermeasures depends on the deviation identification.
Consequences: Low process quality, Process deadlocks, Scrap and rework

Mechanism

Name: Data Collection and Reconciliation.1
Instruction: Sensing, collecting, and analyzing task data. Mapping with expected data and initiating countermeasures, if required.

Solution

Name: Process Data Reconcilitaion.1
Measures: Integrating sensors to collect and analyze process task data. Mapping collected data with expected data to find deviations. Initiating countermeasures, if required.

Value Proposition

Name: Situational Awareness.1
Key Capabilities:
Comprehensive perception
IoT Maturity:
Situational awareness

Technology

Name: Sensing.1
Smart Devices: Activity-aware systems sensing and collecting data.
Data Processing: Processing situational data in edge-devices or (cloud-) servers. Comparing data with defined values/thresholds.

Context

Name: Context.2
Characteristics: Poorly or incorrectly executed tasks must be identified. Appropriate countermeasures must be initiated.

BPI Pattern

Name: Deviation Detection.1
Example: After the filling of toxic gas cylinders, the worker must place the cylinders in the right area for subsequent tasks. If the cylinders are placed in the wrong area, this involves high risk potential, e.g., for placing in food or chemical gas areas. Sensors collect locational data of the toxic cylinders. This data is processed, analyzed, and mapped with expected data of the process task. If the perceived data does not comply with the expected locational data, a deviation is detected. This information can be used to prevent a wrong placement or to initiate a repositioning of the toxic cylinders and a scrapping of the food or medical gases.

Effect

Name: Effect.2
Cost: (0) neutral
Time: (0) neutral
Quality: (+) positive
Flexibility: (0) neutral

Performance Indicator

Name: Quality.2
Performance Measures: Error recognition rate, Rework rate

Process Perspective

Name: Functional.1
Description: Process task monitoring

Process Perspective

Name: Data.2
Description: Collection and reconciliation of data

Figure 7: Deviation Detection pattern.

The exemplary process is taken from the Linde plc, where customers are guided through the purchasing process at a gas vending machine.

This is done be capturing process-related data and highlighting the next process steps via light bars. Another suitable literature application is the training of new employees in a manufacturing company (König, 2019). Employees are guided through tasks by tracking the current process data and visualizing process descriptions of subsequent tasks. Other organizations have implemented applications to guide the employees through production or logistic processes by capturing environmental and process data, processing it, matching it with process models, and providing guidance for tasks (De Vries, 2015).

5.3 Pattern: Deviation Detection

The second pattern *Deviation Detection* is exemplary described using a cylinder filling process of the Linde plc. A main challenge for organizations is the detection of process deviations during runtime to identify incorrect task executions and adequately adapt the subsequent process flows. Deviations lead to low process quality, process deadlocks, or the need for process support. The pattern is illustrated in Figure 7. After the filling of toxic cylinders, they must be placed in the right areas according to the process description. Incorrect task executions include high risk potential. By implementing location sensors that collect data of the task execution and collating it with expected values from the process description, deviations can be detected. This enables the initiation of countermeasures and leads to an improved error recognition rate which has a positive impact on the overall process quality. The pattern addresses the functional and data perspectives, as the execution of the process task is monitored. The IoT technology includes activity-aware smart devices that process situational data on edge devices or (hybrid) cloud servers. To identify deviations of any kind, the key capability comprehensive perception must be exploited enabling situational awareness of all process details. Similar industrial applications ca be found for the detection of machine failures where sensor data is used for diagnostics and detection of deviations, e.g., at leakage detection (Ammirato, 2019) or other anomalies (Schneider, 2019).

5.4 Expert Survey Results

After the practitioners derived six patterns from the sample of 34 applications, they were asked to perform an expert survey. They received a list of six statements for which they needed to indicate their agreement or disagreement. This followed the proven psychometric tool of the Likert scale (Albaum, 1997). The statements were formulated in a way that allows conclusions to be drawn about the three evaluation criteria. Table 1 shows all statements and the obtained survey results. As shown, most of the practitioners agreed or strongly agreed with all statements. Only for the second statement, one practitioner could not specifically state, if the metamodel's degree of abstraction and generalization is appropriate for the derived patterns. Within the statements, especially the first three refer to the criterion *usefulness*, the fourth to the criterion *conciseness*, and the fifth and sixth to the criterion *robustness*. In a subsequent discussion, the experts stated, that the metamodel enabled an appropriate illustration of IoT-based BPI patterns. Also, the classes and attributes supported the analysis of heterogenous IoT applications and the derivation of generic patterns.

Table 1: Expert survey results.

No.	Statement	Strongly agree	Agree	Neutral	Disagree	Strongly disagree
1	The metamodel's classes and attributes enable an appropriate illustration of IoT-based BPI patterns.	29%	71%	0%	0%	0%
2	The classes and attributes allow an appropriate degree of abstraction and generalization.	29%	57%	14%	0%	0%
3	The created patterns can illustrate and describe generic business process problems and potential solutions provided by IoT technology.	14%	86%	0%	0%	0%
4	Extending the metamodel would contradict its generic design, limit its generality, and decrease the number of applications that can be covered by a pattern.	29%	42%	29%	0%	0%
5	Removing classes and attributes would reduce the expressiveness of the patterns.	86%	14%	0%	0%	0%
6	The classes and attributes allow a sufficient differentiation of the represented patterns.	71%	29%	0%	0%	0%

6 CONCLUSION

The contribution of this paper is a metamodel to illustrate generic IoT-based BPI patterns as an extension of the metamodel for BPI patterns of Falk et al. (2013). In two development iterations, additional classes and attributes were discovered and irrelevant ones were dropped. At first, we performed an inductive development iteration including an SLR followed by open and axial coding. Based on the results, additional classes and attributes could be derived and added to the existing metamodel. The first metamodel draft was then refined by conducting a Delphi study with nine experts from industry and academia. To evaluate the final metamodel, seven practitioners from the Linde plc and two other corporations analyzed a set of 34 real-life IoT application of their business areas. Eventually, they derived six IoT-based BPI patterns and illustrated them using the metamodel. In a subsequent survey the experts assessed the metamodel according to the predefined evaluation criteria *usefulness*, *conciseness*, and *robustness*. The survey showed that the metamodel sufficiently meets these criteria. Despite the rigorous research methodology, the contribution is not without limitations due to the nature of DSR. Following an inductive approach for metamodeling is a proven concept that provides several advantages arising from building up on actual observations. However, the underlying SLR cannot cover all existing data of the phenomenon under investigation. The identification of literature is limited to the incorporated databases and formulated queries. To mitigate this subjectivity, we conducted a subsequent Delphi study. This enabled both a formative evaluation of the first metamodel draft and the inclusion of broad expert knowledge.

Further research should be conducted in various directions. Having evaluated the usefulness, conciseness, and robustness of the metamodel, the actual *applicability* of the generated patterns must be assessed. In this regard, we plan to create and introduce patterns to industrial organizations. This enables a further evaluation, if the patterns can be effectively used to realize IoT-based BPI applications. Moreover, the creation of a comprehensive pattern catalogue would provide additional benefit and validate the metamodel itself.

REFERENCES

Albaum, G., 1997. The Likert Scale Revisited. *Int. J. Mark. Res. 39(2)*.

Alexander, C., 1977. *A pattern language: towns, buildings, construction*. Oxford University Press, New York City.

Ammirato, A., Sofo, F., Felicetti, A. M., Raso, C. 2019. The potential of IoT in redesigning the bank branch protection system: An Italian case study. *Bus. Process Manag. J. 26(7)*.

Atzori, L., Ier,a A., Morabito, G., 2010. The Internet of Things: A survey. *Comput. Netw. 54(15)*.

Bosche, A., Crawford, D., et al., 2016. How providers can succeed in the Internet of Things. *Bain & Company Inc.* https://www.bain.com/insights/how-providers-can-succeed-in-the-internet-of-things/.

Corbin, J. M., Strauss, A., 1990. Grounded theory research: Procedures, canons, and evaluative criteria. *Qual. Sociol. 13*.

Coskun, S., Basligil, H., Baracli, H., 2008. A weakness determination and analysis model for business process improvement. *Bus. Process Manag. J. 14(2)*.

De Vries, J., de Koster, R., Stam, D., 2015. Exploring the role of picker personality in predicting picking performance with pick by voice, pick to light and RF-terminal picking. *Int. J. Prod. Res. 54(8)*.

Del Giudice, M., 2016. Discovering the Internet of Things (IoT) within the business process management: a literature review on technological revitalization. *Bus. Process Manag. J. 22(2)*.

Dumas, M., La Rosa, M., Mendling, M., Reijers, H. A., 2018. *Fundamentals of Business Process Management*. Springer Verlag, Berlin, Heidelberg.

Falk, T., Griesberger, P., Johannsen, F., Leist, S., 2013. Patterns for Business Process Improvement - A first approach. *In 21st European Conference on Information Systems. ECIS*.

Ferstl, O. K., Sinz, E. J., 2013. *Grundlagen der Wirtschaftsinformatik*. Oldenbourg Verlag, München.

Forster, F., 2006. The idea behind Business Process Improvement: Toward a Business Process Improvement pattern framework. *BP Trends 4*.

Fowler, M., 1996. *Analysis Patterns: Reusable object models*. Addison-Wesley, Boston.

Fowler, M., Scott, K., 1997. *UML Distilled: Applying the standard object modelling language*. Addison-Wesley Longman, Amsterdam.

Gamma, E., Helm, R., Johnson, R., 1994. *Design Patterns: Elements of reusable object-oriented software*. Prentice Hall, Hoboken.

Gonzalez-Perez, C., Henderson-Sellers, B., 2008. *Metamodelling for software engineering*. Wiley, Hoboken.

Hempel, C. G., 1966. *Philosophy of natural science*. Prentice-Hall, Upper Saddle River.

Hevner, A. R., March, S. T., Park, J., and Ram, S., 2004. DS in information systems research. *MISQ 28(1)*.

Hohpe, G., Woolf, B., 2003. *Enterprise Integration Patterns: Designing, building, and deploying messaging solutions*. Addison Wesley, Boston.

Jablonski, S., Bussler, C., 1996. *Workflow Management: Modeling concepts, architecture and implementation*. Cengage Learning EMEA, Andover.

Janiesch, C., Koschmider, A., et al., 2020. The Internet of Things meets Business Process Management: A manifesto. *IEEE Trans. Syst. Man. Cybern. Syst. 6(4)*.

Kortuem, G., Kawsar, F., Sundramoorthy, V., Fitton, D., 2010. Smart objects as building blocks for the Internet of Things. *IEEE Internet Comput. 14(1)*.

König, U. M, Röglinger, M., Urbach, N., 2019. Industrie 4.0 in kleinen und mittleren Unternehmen – Welche Potenziale lassen sich mit smarten Geräten in der Produktion heben? *HMD der Wirtschaftsinformatik 56*.

Kühn, H., Karagiannis, D., 2005. *Strategie-, Prozess- und IT-Management: Ein Pattern-orientierter Integrationsansatz*. In Ferstl, O. K., Sinz, E. J., et al. (ed) Wirtschaftsinformatik 2005. Physica, Heidelberg.

Li, Y., Hou, M., Liu, H., Liu, Y., 2012. Towards a theoretical framework of strategic decision, supporting capability and information sharing under the context of Internet of Things. *J. Inf. Technol. Manag. 13*.

Liu, V., 2017. Business benefits of the Internet of Things: Gartner report. *Gartner Inc.* https://www.gartner.com/en/documents/3806366/business-benefits-of-the-internet-of-things-a-gartner-tr.

Loo, R., 2002. The Delphi method: a powerful tool for strategic management. *Policing 25(4)*.

Minerva, R., Biru, A., Rotondi, D., 2015. Towards a definition of the Internet of Things (IoT). *IEEE*. https://iot.ieee.org/definition.html.

Ng, I., Wakenshaw, S., 2017. The Internet-of-Things: Review and research directions. *Int. Res. Mark. 34(1)*.

Patterson, R. E., 2017. Intuitive cognition and models of human-automation interaction. *Hum Factors 59(1)*.

Peffers, K., Tuunanen, T., Rothenberger, M., Chatterjee, S., 2007. A DSR Methodology for Information Systems Research. *J. Manag. Inf. Syst. 24(3)*.

Prat, N., Wattiau, I., Akoka, J., 2015. A Taxonomy of Evaluation Methods for Information Systems Artifacts. *J. Manag. Inf. Syst. 32(3)*.

Reijers, H. A., Liman Mansar, S., 2005. Best practices in business process redesign: an overview and qualitative evaluation of successful heuristics. *Omega 33(4)*.

Reinfurt, L., Breitenbücher, U., Falkenthal, M., et al., 2016. Internet of Things patterns. In *21st European Conference on Pattern Languages of Programs*.

Schmidt, D. C., Stal, M., Rohnert, H., Buschmann, F., 2000. *Pattern-oriented software architecture*. Wiley, Hoboken

Schneider M., Lucke D., Adolf T., 2019. A cyber-physical failure management system for smart factories. *Procedia CIRP 81*.

Serral, E., Stede, C. V., Hasić, F., 2020. Leveraging IoT in etail Industry: A Maturity Model. In *22nd Conference on Business Informatics*.

Sethi, P., Sarangi, S. R., 2017. Internet of Things: Architectures, protocols, and applications. *Int. J. Electr. Comput. Eng. 2017*.

Skaržauskienė, A., Kalinauskas, M., 2015. The internet of things: when reality meets expectations. *Int. J. Innov. Learn. 17(2)*.

Stoiber, C., Schönig, S., 2021. Process-aware Decision Support Model for Integrating Internet of Things Applications using AHP. In *23rd International Conference on Enterprise Information Systems. ICEIS*.

Tai Angus Lai, C., Jackson, P. R., Jiang, W., 2018. Designing service business models of the IoT: Aspects from manufacturing firms. *Am. J. Math. Ma. Sci. 3(2)*.

Templier, M., Paré, G., 2018. Transparency in literature reviews: an assessment of reporting practices across review types and genres. *Eur. J. Inf. Syst 27(5)*.

Venable, J., Pries-Heje, J., Baskerville, R., 2016. FEDS: a Framework for Evaluation in Design Science Research. *Eur. J. Inf. Syst. 25*.

Vom Brocke, J., Niehaves, A., et al., 2009. Reconstructing the giant: On the importance of rigour in documenting the literature search process. In *17th European Conference on Information Systems, ECIS*.

Washizaki H., Ogata S., Hazeyama A., et al., 2020. Landscape of architecture and design patterns for IoT systems. *IEEE Internet Things J.*

Winter, R., vom Brocke, J., et al., 2009. Patterns in business and information systems engineering. *Bus. Inf. Syst. Eng. 1(6)*.