Testing Technical Feasibility in CPS Development Projects

Moritz von Hoffen[1], Friedrich Chasin[1], Martin Matzner[2], Florian Plenter[1], and Jan H. Betzing[1]

[1] University of Münster – ERCIS, Leonardo-Campus 3, 48149 Münster, Germany,
{moritz.von.hoffen,friedrich.chasin,
florian.plenter,jan.betzing}@uni-muenster.ercis.de
[2] University of Erlangen-Nürnberg, Digital Industrial Service Systems, Nuremberg, Germany
martin.matzner@fau.de

Abstract. Cyber-physical systems (CPSs) are service systems that connect a product's physical and computational elements through telecommunication networks. Typically, the processes in CPSs are executed on this physical and computational infrastructure. As the developing of new CPS is costly, testing and validating a CPS's design at an early stage of development is desirable in order to avoid potential bad investments. The high development and potentially high hardware costs, however, make it difficult to create a full CPS prototype only for testing. This work uses Trkman's critical success factors of business process management (BPM) as a theoretical lens and identifies "technical-feasibility fit" as an additional complementary success factor. Based on these factors, we develop a method for creating CPS testbeds that allow testing of CPSs at lower costs at an early stage of the development. We demonstrate the method's application by a case in which we develop a testbed for an electric vehicle charging service.

Keywords: Cyber-physical Systems, Critical Success Factors, Prototyping

13th International Conference on Wirtschaftsinformatik,
Februrary 12-15, 2017, St. Gallen, Switzerland

von Hoffen, M.; Chasin, F.; Matzner, M.; Plenter, F.; Betzing, J. H. (2017): Testing Technical Feasibility in CPS Development Projects, in Leimeister, J.M.; Brenner, W. (Hrsg.): Proceedings der 13. Internationalen Tagung Wirtschaftsinformatik (WI 2017), St. Gallen, S. 589-603

1 Introduction

Service is the value created in relational interaction processes [1] that connect a company to several "collaborators" [2, p. 492] such as partners, employees, and suppliers. All entities together form the *service system*, which VARGO and LUSCH defined as a dynamic configuration of four types of resources, i.e., people, technologies, organization, and information [3]. *Service Development* refers to a firm's approach to creating new service offerings and has been described as a cyclical process that includes various planning and implementation activities at the progressive stages of "Design", "Analysis", "Development", and "Launch" [4]. Notably, a service development project can also return to earlier stages if later planning and development activities require modifications of the service concept. Against this background, service development requires that service concepts are tested repeatedly for their business value and for their operational feasibility [5]. For instance, shortly after initial idea generation, firms typically evaluate the service concept ideas through a "screening" [6]. However, the individual development stage activities related to the design of the service, processes, and the actual system require more exhaustive testing [4]. Business value embodied in, e.g., profitability, growth and reward potential, as well as competitive advantage [6, 7] is typically assessed using conventional qualitative and quantitative market research techniques like surveys, focus groups, one-on-one interviews and conjoint analysis [8].

In contrast, the testing of the operational feasibility of a service concept requires to look deeply into the service system's value creation processes as well as the technological and informational resources they use. "Prototyping" is one approach to achieve rapid customer-centric service experimentation [9]. In this context, it is an important question how *service prototyping* can be used to "materialize an integrated set of service system components, such as the people, the process, the technology, and the physical evidence" [10, p. 137]. According to OSTROM et al., prototyping has not received sufficient attention in service research, and thus, they feature service prototyping in their recent list of important service research areas [10].

Especially in the domain of cyber-physical systems (CPSs), technology components are of particular significance. Typically, CPSs connect (remote) computational and physical entities, e.g., sensors and actuators, via global computational networks [11]. In this context, *prototyping* is highly important for this type of service system because of the requirement that complex technical infrastructures have to be built at early stages of the service development—even before the progress and the processes can be fully tested.

Against this backdrop, the present paper addresses the following research question: *How can service prototyping materialize the process and technology components of cyber-physical systems?* The contribution of this paper lies in the design of a method for creating CPS testbeds. We intend to improve CPS service development by facilitating prototyping and testing for operational feasibility at early stages of the development process and at reasonable costs.

The development of the method is informed through the theory of *task-technology fit (TTF),* a theoretical lens that has been applied previously in business process management (BPM) [12, 13]. The TTF theory helps to assess whether certain technologies are appropriate to a given process. Therefore, this paper also seeks to synthesize research on prototyping in service development and research on success factors in BPM, which have so far been considered only separately.

The remainder of this article is structured as follows: The next section gives background on the testing of CPS with regard to the process perspective and success factors. Then we explain our research approach, followed by the method and a demonstration of its application in a project that develops a service for electric vehicle (EV) charging. The evaluation section provides first evidence of the method's usefulness. A discussion of our results follows, and the final section concludes the article.

2 Research Background

2.1 Challenges in Testing CPS

CPSs are specific service systems including networked computational systems that are partly embedded into physical objects [11]. Sensors and actuators connect the physical and digital worlds. An ever-growing number of CPSs, which have become ubiquitous in every-day life, generate a vast amount of data, with typical applications ranging from smart grids [14], physical infrastructures in transportation [15], traffic and process control to automotive and medical systems [16].

CPSs are complex systems with complex processes that typically run on expensive hardware. In particular, the embedding of physical components requires higher standards for reliability and safety as system failures can result in severe damages, e.g., of the environment [17]. Embedded systems such as driving assistance or brake control are examples of CPSs integrated in every-day systems, whose failures can result in serious consequences for the public. Moreover, the behavior of CPSs cannot always be predicted.

While traditional end-to-end business processes are implemented within or across a few application systems, processes in CPSs add another layer of complexity. In effect, parts of the business logic are shifted into these embedded systems [18]. From the business perspective, addressing the challenges posed by the nature of CPSs requires considerable investment. Failing in the latter stages of the development due to miss-specified processes that are unable to execute within and across the CPS can be costly, so guarding against such situations is critical for managing and executing business processes.

2.2 Process-Focus in Service Design and Testing

Testing CPS for operational feasibility is of great importance throughout the various stages of service development [5]. Prototyping has been discussed as a promising ap-

-proach to achieve a balance between receiving early insight on the feasibility and the costs associated with the testing activities [10]. The key intellectual challenge in service prototyping is to achieve an integrated service experimentation that materializes all relevant components of the service systems in a way so that the service stakeholder can make sense of the service and make reasonable decisions about the progress of the development project [9]. The scope of this paper has been set to processes and technology, which are the most significant components of CPS service systems.

The BPM literature has put forth constructs, models, and theories that help to study the relationship between processes and technology. Notably, in an attempt to identify critical success factors (CSFs) for BPM, TRKMAN demands "continuous improvement efforts" for BPM and two types of "fit" for business processes [12, p. 126]. The "fit between business environment and business processes" has been explained by the *contingency theory* [19], which in essence states that there exists no universal or "best way" to manage an organization. Instead, achieving an appropriate organization is contingent to various internal and external constraints. Accordingly, business processes have to be designed so that they meet the constraints of the process environment. The "fit between business processes and technology" [12, p. 127] has been explained through *task-technology fit (TTF)*—a theory that identifies that a positive impact of information technology (IT) investments on organizational performance is subject to matching IT and business processes.

The need for "dynamic improvement" of business processes is justified by the theory of *dynamic capabilities*, which postulates that organizations need to address changing environments through the ability to integrate, build, and reconfigure internal and external competences. Therefore, business processes need to be reviewed for both types of fit continuously.

We focus on TTF, which provides means to study the CPS's *process* and *technology* service components in conjunction. While testing in software development projects already accounts for about one third of development cost [20], the testing and validation of the distributed and embedded components of a CPS is even more complex and costly [21] and thus underlines the importance of a proper TTF.

3 Research Approach

To approach the problem of testing the operational feasibility of a service concept in the context of a CPS throughout different stages of the service development process, we perform two research activities: At first, we aim at the derivation of a framework for critical success factors in BPM from the extant literature. This step is required to examine and categorize different state-of-the-art CSFs to identify the gap and motivate the extension of the framework with an additional CSF of testing the operational feasibility of the service concept. The second strand of our research deals with the development of a method that is capable of closing the identified gap.

Table 1. Research Steps

#	Step	Activities	Outcomes
1	*Analyze CSFs*	Investigate extant CSFs Align CSFs and BPM Analyze applicability in use case scenario	Framework Identified gap
2	*Propose Extension*	Propose testbedding as an extension to the TTF	Motivation Proposed extension
3	*Method Design*	Design the method Apecify activities	Method instantiation
4	*Demonstration*	Demonstrate the extension Iterative execution Adopt continuous improvement	Evaluation

Table 1 summarizes the steps undertaken in this research and the generated outcomes. In the first step, we identify relevant CSFs in the extant literature (see CSFs labelled with [*] in Table 2). Taking the specific requirements imposed on service testing in the CPS context, we propose the extension of the general framework of success factors in BPM with an additional CSF (see CSF_{2+} in Table 2). We then design the method suitable for ensuring and warranting that the chosen technology set for the service at hand aligns with the corresponding business processes. Finally, we complement the demonstration of the method with a discussion of the proposed approach against related testing procedures.

Table 2. Overview of *Critical Success Factors* ([*] according to [12])

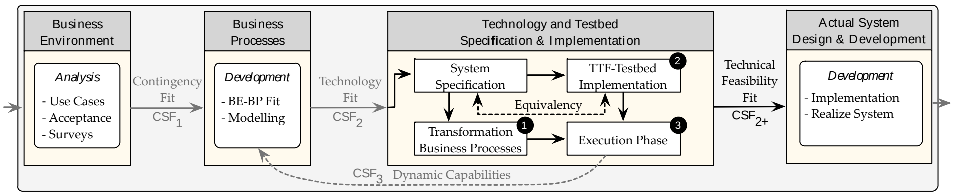
CSF_1[⋆]: *Contingency Fit*		
Activities	***Outcomes***	***Literature***
•Evaluation of business engineering •Alignment of strategy with business engineering •Conceptualization and modeling of business processes	Contingency of the business engineering and business processes	[12, 13, 19, 22]
CSF_2[⋆]: *Task-Technology Fit*		
Activities	***Outcomes***	***Literature***
•Alignment of supporting IT with business processes	Technological compatibility of IT and business processes	[12]
CSF_{2+}: *Technical Feasibility Fit*		
Activities	***Outcomes***	***Literature***
•Mapping of business processes to a state-based representation •Identify the technologies required for testbed implementation •Assemble the testbed •Execution of business processes in testbed	Feasibility of selected technological components	—
CSF_3[⋆]: *Dynamic Capabilities*		
Activities	***Outcomes***	***Literature***
•Continuous improvement of business processes •Responding to changes in the business engineering	Mature business processes achieved through continuous improvement	[12, 19]

4 A Method for Creating CPS Testbeds

Against the backdrop of high risks associated with business processes relying on CPSs, we enhance TRKMAN's model of CSFs [12] by an additional CSF that follows the TTF. We introduce the specification and implementation of a testbed as means of ensuring the *technical feasibility fit* between the chosen technology set and the business processes (cf. Figure 1). In effect, a testbed combines virtual, simulated, and physical components into a configurable experimental setup for testing [23]. In an ideal world, the behavior and properties of the testbed are equivalent to ones of the specified service system. Thus, we use the term *testbed equivalency*.

Our work aims at achieving an optimal TTF with a technology ensemble feasible to execute the business processes, while treating the remaining activities required to address further CSFs as a *black box*. Assessing the TTF can be expensive, which is especially true for distributed processes that run on heterogeneous and specialized hardware. Prototype development with a testbed combines the benefits of early testing and validation with cost savings, because the actual hardware roll-out can be postponed until the testbed has been used to validate the correct execution of all involved (business) processes. Hence, prototyping can "reduce the chances of costly new service failures" [10, p. 137].

Testbeds have to correctly imitate the execution of the business processes, and thus, require a precise specification of the target system's behavior. We therefore limit the scope to business processes that use standardized and established technologies, techniques and protocols, so that the behavior of the system can be anticipated.



4.1 Steps of the Method

Figure 1. Embedding of Testbedding into the Framework of CSFs for BPM

Figure 1 locates the proposed *technology and testbed specification and implementation* within the process and framework of CSFs in BPM. The activity blends in after CSF_1 and CSF_2 have been achieved through the contingency and technology fits. At this point, the business processes have been modelled and formalized. Based on the business processes and underlying standards, a set of technologies, i.e. software and hardware components, has been chosen. The testbed method consists of three steps: First, the resulting business processes must be *transformed (1)* into a state-based representation. Simultaneously, the *testbed implementation (2)* is performed. The testbed equivalency to the actual system must be assured through the equivalency of the testbed specification to the system specification. Subsequently, the implemented and configured testbed is put to use in the *execution phase* (3) by executing the transformed business processes from step (2).

(1) Transformation of Business Processes into a State-based Representation: Business processes are typically represented as models and the Event-driven Process Chain (EPC) and Business Process Model and Notation (BPMN) are arguably the most prominent graphical process modeling languages in both, academia and practice. To ensure correct process execution and *soundness*, one needs to transform the EPC or BPMN models into a *state-based* representation. The utilization of a state-based representation allows to precisely comprehend the current state of the process and check every state transition for compliance. In this context, *Petri nets* are recommendable [24] which is justified by the large existing body of knowledge on formal validation of Petri nets [25]. Moreover, for the actual transformation, one can make use of existing and well-tried methods for *model-to-model transformation* to convert the business processes into Petri nets. The mapping itself is straightforward: "*tasks* are modeled by *transitions*, *conditions* are modeled by *places*, and *cases* are modeled by *tokens*." [26, p. 15]. A *marking* can be understood as a snapshot that reflects the Petri net's state at a certain point in time. In order to make the state transitions of the resulting Petri net transparent, the individual states must be represented in such a way that they are *observable*. This approach is similar to *lean manufacturing* or *Andon systems* where certain situations are signaled. Several options are conceivable for providing such an output like displays, acoustic signals, and light-emitting diodes (LEDs). Once a suitable and observable state representation has been decided upon (e.g., LEDs), a coherent mapping of the individual Petri net markings, i.e., states, must be developed. A naïve solution is to assign an individual LED to each place on the Petri net, which would visualize the presence of a token at the corresponding place. However, the number of places in the resulting Petri net can be large for complex business processes. This can be mitigated by using different states of the same signal emitter to code the marking, e.g., using multi-colored LEDs and modes like *on/off* or *blinking/pulsing* in different intervals. A display can also be attached that can be used to output the state as well as accompanying information such as enabled transitions or a history of states, which can be used for *backtracking* purposes to achieve full coverage of the process.

(2) Testbed Implementation: Based on the results of the TTF assessment, a testbed has to be specified that ensures *equivalency* to the technology set intended for the implementation of the productive IT infrastructure. Due to heterogeneity in required capabilities among various use cases and domains, the hardware selection process needs to be considered individually. However, a careful evaluation of the underlying task-technology fit is mandatory to provide a tangible basis for choosing suitable hardware components for the intended testbed. Naturally, the selected hardware should be capable to imitate the actual productive component of the CPS.

Important activities in this development stage include the assessment and comparison of different hardware and software vendors and sources, as well as their compatibility. Active support and maintenance should be taken into account as well. The decision regarding a suitable means for the output depends on many factors like the total number of devices in the testbed, the processes' complexity, and the degree of concurrency. Depending on which output mean(s) is/are chosen, a formal mapping and representation of the different states must be defined. Finally, the testbed device(s) is/are assembled and programmed so that it can emulate the intended business processes.

(3) Execution Phase: The set of business processes to be tested should be compiled beforehand and in accordance with the testbed specification. The resulting *test suite* is then processed by executing the different processes in the testbed environment. In the spirit of *continuous improvement*, the processes are executed iteratively within the testbed. If an abnormality is experienced during the execution phase, this information is recorded and re-evaluated in an iterative manner in a subsequent execution round. Each execution of the processes in the testbed environment provides feedback to the specification phase until the result meets the acceptance criteria. In some cases, the testbed might also prove that a given business process is unfeasible for real-world execution. This information flow and the subsequent addressing of defects results in a demonstrated *technical feasibility* of the technology—given the assumption that the testbed and the final system are equivalent when the processes are emulated. This additional step that contemplates the initial TTF constitutes an additional CSF: *Technical Feasibility Fit*. Once the "sweet spot" in terms of robustness has been reached, the replacement of the testbed with the actual production system is approached.

5 Demonstration

5.1 Project Setting: EV Charging Infrastructure

We applied the testbed method in the domain of EV charging. EVs are charged using charging points, which combine electrical and computational components. The CPS at hand comprises a charging infrastructure of networked charging points and an information system (IS) that, among other tasks, controls the individual charging points, authorizes users to unlock a charging point and charge their vehicles, and handles the billing of charging transactions.

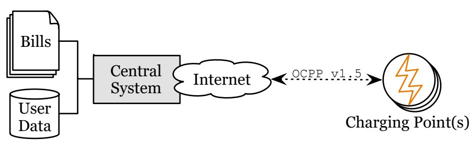


Figure 2. Central System and Charging Points

In particular, the processes for controlling the charging infrastructure, which comprises the *charging points* and a corresponding *central system* (see Figure 2), have been formalized in the *Open Charge Point Protocol (OCPP)*. The OCPP represents a *de facto standard* and protocol for the communication between the central system and the individual charging points [27]. The communication is realized by sending OCPP-based SOAP requests over HTTP. According to the OCPP, charging points can be in one of four states and different messages are used to either initiate or communicate a change of the state as visualized in Figure 3. For instance, an *expired reservation* is to be detected by the charging station itself which will then change its status to *available* whereas a request to *cancel a reservation* is sent by the central system to a specific charging station.

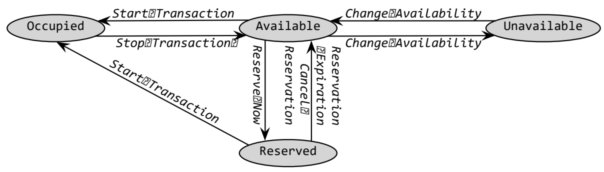


Figure 3. Transition System of a Charging Point According to the *OCPP v1.5*

5.2 Method Application

We perform the three aforementioned steps of the method (cf. Figure 1):

(1) Transformation of Business Processes into a State-based Representation: As the testbed is supposed to ensure that the business processes and the real world are compatible, a transformation into a state-based representation is performed. This is realized by transforming the individual BPMN models into a Petri net representation. In our case, we transformed the business processes that have been specified for the central system and a charging station (cf. Figure 2). To make the different states "experiencable" and observable, we relied on signaling using LED states to represent the states of the charging point:

$$\text{State} \equiv \{[LED_{red} = x_1], [LED_{yellow} = x_2], [LED_{green} = x_3]\},\ x_i \in \{\text{off}, \text{on}, \text{pulse}, \text{blink}\} \quad (1)$$

Figure 4 illustrates how the individual states of the *BPMN model* are mapped to a *Petri net* and specific *LED states*. Each marking of the resulting Petri net corresponds to a unique LED allocation.

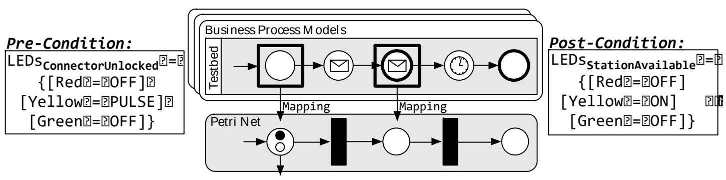


Figure 4. *Mapping* of BPMN to *Petri Net* and *LED States*

(2) Testbed Implementation: The architecture of charging stations mandates several functional requirements for the testbed implementations: reading of NFC cards, mobile

Internet connectivity, and exchange of messages with the central system based on OCPP v1.5. The publicly available Web Service Description Language (WSDL)[1] files were used for the specification and software development for the testbed and the corresponding central system. We then engineered a testbed device to resemble a charging station for the testbedding.

LEDs indicate the state of charging station as in Figure 3 (i.e., idle, reserved, out-of-business, connector (un-)locked, charging, charging finished. In addition to the LED signals, the transition sequences are logged and shown on a display. Because of its low cost, versatility, broad support, and active community, a *RaspberryPi* Model B[2] served as the basis for the development of the testbed .

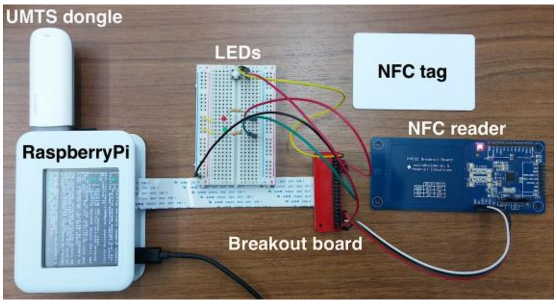


Figure 5. The *Testbed* Devices

Figure 5 shows the testbed device imitating a charging station including a near field communication (NFC) reader and the connected LEDs. The device itself is mounted in a plastic housing and also features a touchscreen display. The device connects to the Internet via universal mobile telecommunications system (UMTS) or wireless local area network (WLAN). Finally, the previously developed software was deployed on the testbed device.

(3) Testbed Execution: After specifying the mappings between the states of the business process and the states of the LEDs, the testbed is used to execute the processes. A continuous improvement cycle is included in the execution. All methods in the standard were tested against a OCPP-compliant central system. Correct business process execution is evaluated for each execution round by observing the specific outputs—that is, the messages and LED states—and comparing them to the expected output. In order to comprehend the correct execution for each state, the processes can be executed step-wise (comparable to debugging a software implementation). Traces of the execution of the business processes are logged for later analysis.

The testbed allowed us to test different processes that could otherwise not have been tested by typical means of simulation. The ability to test a transaction from the start to the finish by holding a NFC tag in front of the reader of the testbed device helped us to come up with a robust solution. Errors found in an early stage could already be ad-

[1] OCPP v1.5 WSDL - http://www.openchargealliance.org/?q=node/9

[2] RaspberryPi Model B - https://www.raspberrypi.org/products/model-b/

dressed within the next iteration (cf. CSF_3). Throughout the development phase, the testbed was used to validate the correct execution according to the OCPP. Thus, all components comprising an EV-charging infrastructure could be executed and tested in a realistic setting.

6 Discussion

6.1 Contributions to the Practice of New CPS Development

We proposed to extend the task-technology-fit perspective towards the testing of technical feasibility in the development of new CPSs, which led to the identification of "Technical Feasibility" as an *additional* critical success factor (CSF) for such projects. Motivated by the observation of OSTROM et al. that the question how service prototyping can "materialize an integrated set of service system components" [10, p. 137] has not been sufficiently addressed by service research, this paper provides an illustrative example of creating a testbed to check a CPS's technical feasibility at an early stage of service development. The testbed mimics the CPS and allows to check the CPS's business processes for technical feasibility and correct execution. We believe that this paper contributes meaningfully to the practice of new CPS development, because it shows a way to test a CPS before implementing the "real" technical components of the CPS to the full extent. Especially in highly standardized environments, our approach may enable service, product and IT engineers to align their designs not only with the business envi- ronment but also with the enabling hardware. From a managerial perspective, taking the CSF of "Technical Feasibility Fit" carefully into account can mitigate the risk and thereby the cost of erroneous specifications that would surface late in the development process. A testbed mimics the specified system and makes it accessible through a hands-on method, which enables developers to detect such erroneous specifications earlier. In regard to the demonstrated EV-charging infrastructure testbed, announced revisions of the OCPP will be addressed. The IS at hand implements business processes that have to be executed consistently across a fleet of heterogeneous charging points (e.g., different vendors, models, and revisions). Inter-compatibility can be assured using the presented method.

6.2 Contributions to Service Research

When proposing a new method, there is a need to demonstrate its "worthiness" against the existing body of knowledge. We therefore subsequently review the extant literature to demonstrate that this research creates novel scientific knowledge if transferred to contexts other than its originating one. We consider service testing in four streams in the academic literature, viz., (a) *product-service systems*, (b) *cyber-physical systems*, (c) *service marketing* and (d) *service blueprinting* as part of *service engineering*.

Cyber-physical Systems: service testing in CPS literature is mainly interpreted from a computer science point-of-view as the problem to prove the security, privacy, reliability, or resilience at the intersection of embedded computing components and cyber infrastructures [28, 29]. Thus, researchers suggest to rep- resent CPSs using formal specifications, which facilitate the adoption of formal verification techniques [29] as testing means. Formal specifications naturally focus on a CPS's physical and computational aspects.

While the CPS literature addresses interactions between physical and com- putational components only, our work also includes human interactions with physical components during the assessment of technical feasibility. Therefore, unlike the prevalent literature on testing in CPS, our work considers the entire service system comprising of people, technology, organization and information if all the CSFs are taken into account during service development.

Product-service Systems: PSS originates from manufacturing and industrial engineering with a focus on how to develop marketable customer solutions that involve physical components. PSS endeavors are often based on conventional product design processes. The properties relevant for "testing" include product-related quality issues as well as specific economic and environmental benefits of PSS [30]. While many successful PSS implementations have been reported—such as the Electrolux case study of in-flight services [31]—testing is mentioned rarely. Testing PSS primarily focuses on the physical PSS components and provides ways to step-wisely develop service concepts during iterative product design cycles [32].

Service Marketing: the prevalent metrics for assessing service concepts in service marketing relate to financial performance, anticipated market impact [33] or anticipated customer satisfaction [34]. Related to the latter, recent publications suggest techniques that strive to make a future customer's service experiencable at an early stage of service development, such as the customer journey, touchpoint approaches, and storytelling [8]. Roleplaying, design scenarios, story- boards, desktop walk-through, and service staging extend this list [35]. Addition- ally, visualizing techniques such as flow-charting, service blueprints, and process- chain-network diagrams are frequently suggested [36]. "Prototyping", however, has been widely neglected in the service marketing literature [37]. BOWERS early assumed the root-cause in arising cost for people and equipment if one wanted "to create a whole process just for testing" [38]. In the CPS setting, testing procedures from service marketing are not capable of addressing technical feasibility within a service as proposed in this study.

Service Blueprinting: the concept service blueprinting, originally proposed by SHOSTACK, is widely used in practice to analyze and design customer interaction in service systems [39]. The service blueprint depicts the division and visibility of a service system's work, structured by the actors (customers and providers) and stages (front-stage and back-stage) [40]. Blueprinting has been reported to be beneficial in new service development, management and control of existing service processes, and customer preferences monitoring (see [41]). It explicitly shows the physical evidence that is seen by the customer during various stages of service delivery. While testing back-stage activities by means of a service blueprint comes closest to what our method intends to achieve, blueprinting focuses on the non-IT components of the service system. This is why we also failed to find guidance to our problem of testing a CPS's technical feasibility in this stream of literature.

6.3 Limitations

In this paper, we proposed an enhancement to the CSFs for BPM in the context of developing new services that are enacted using CPSs. Currently, the scope of the testbed method is very limited as it is only applicable if the system behavior can be anticipated or if it is prescribed by a standard. In absence of established and validated standards, a sound testbed specification that is equivalent to the original technology is almost impossible to achieve. Therefore, future research must explicate for which scenarios the approach is suitable as our findings are derived from a very specific use case that basically dictates the technology to be used and thus cannot be generalized. Furthermore, the scope of this research is on technical feasibility. Future research should also focus on an economic evaluation of the testbed method, i.e., introduce key performance indicators that give evidence on cost savings, improved quality of business processes, and gains for the latter phases in CPS development.

7 Conclusion

High development costs make it difficult to create a full CPS prototype only for testing. However, testing is important to ensure the operational feasibility of the CPS design at an early development stage. Our work applied and extended task-technology fit in order to develop a method for creating CPS testbeds. A testbed allows to validate the correct execution of business processes while also providing evidence on the interaction of customers with the physical components. Our major contribution lays in the identification of the technological feasibility as an additional critical success factor in new CPS development, which, if considered carefully, can help to mitigate the risks of premature failure and may save costs of the CPS development and testing. We incorporated technical feasibility into a method for creating testbeds that is applicable when the behavior of the system is prescribed in standards. This method proved to be useful for our purposes in developing a new CPS for EV charging.

References

1. Tuli, K.R., Kohli, A.K., Bharadwaj, S.G.: Rethinking Customer Solutions: From Product Bundles to Relational Processes. Journal of Marketing 71(3), 1–17 (2007)
2. Spohrer, J., Kwan, S.K., Fisk, R.P.: Marketing: A Service Science and Arts Perspective. Edward Elgar Publishing (2014)
3. Vargo, S.L., Lusch, R.F.: Service-Dominant Logic: Continuing the Evolution. Journal of the Academy of Marketing Science 36(1), 1–10 (2008)

4. Johnson, S.P., Menor, L.J., Roth, A.V., Chase, R.B.: A Critical Evaluation of the New Service Development Process: Integrating Service Innovation and Service Design. Sage (2000)
5. Zeithaml, V.A., Bitner, M.J., Gremler, D.D.: Services Marketing: Integrating Customer Focus Across the Firm. McGraw-Hill, New York, NY (2012)
6. Kelly, D., Storey, C.: New Service Development: Initiation Strategies. International Journal of Service Industry Management 11(1), 45–62 (2000)
7. Cheng, C.C.: Market-Creating Service Innovation: Verification and its Associations with New Service Development and Customer Involvement. Journal of Services Marketing 26(6), 444–457 (2012)
8. Zomerdijk, L.G., Voss, C.A.: NSD Processes and Practices in Experiential Services. Journal of Product Innovation Management 28(1), 63–80 (2011)
9. Meiren, T., Burger, T.: Testing of Service Concepts. The Service Industries Journal 30 (4), 621–632 (2010)
10. Ostrom, A.L., Parasuraman, A., Bowen, D.E., Patricio, L., Voss, C.A.: Service Research Priorities in a Rapidly Changing Context. J Service Research 18(2), 127–159 (2015)
11. Geisberger, E., Cengarle, M.V., Keil, P., Niehaus, J., Thiel, C., Thönnißen Fries, H.J.: Cyber-Physical Systems: Driving Force for Innovation in Mobility, Health, Energy and Production. Springer-Verlag (2011)
12. Trkman, P.: The Critical Success Factors of Business Process Management. International Journal of Information Management 30(2), 125–134 (2010)
13. Goodhue, D.L., Thompson, R.L., Goodhue, B.D.L.: Task-Technology Fit and Individual Performance. MIS Quarterly 19(2), 213–236 (2014)
14. Sridhar, S., Hahn, A., Govindarasu, M.: Cyber-physical system security for the electric power grid. Proceedings of the IEEE 100(1), 210–224 (2012)
15. Bradley, J.M., Atkins, E.M.: Optimization and control of cyber-physical vehicle systems. Sensors 15(9), 23020–23049 (2015)
16. Mitchell, R., Chen, I.R.: Behavior Rule Specification-based Intrusion Detection for Safety Critical Medical Cyber Physical Systems. IEEE Transactions on Dependable and Secure Computing 12(1), 1–1 (2014)
17. Knight, J.C.: Safety critical systems: challenges and directions. In: Proc ICSE 2002. pp. 547–550 (2002)
18. National Academy of Science and Engineering: Cyber-Physical Systems - Driving Force for Innovations in Mobility, Health, Energy and Production. Tech. rep., acatech, Berlin, Germany (2011)
19. Hong, K.K., Kim, Y.G.: The critical success factors for ERP implementation: an organizational fit perspective. Information & Management 40(1), 25–40 (2002)
20. Tassey, G.: The Economic Impacts of Inadequate Infrastructure for Software Testing. NIST, RTI Project 7007(011), 309 (2002)
21. Lee, E.A.: Cyber Physical Systems: Design Challenges. In: Proc ISORC '08. pp. 363–369. Orlanda, FL (2008)

22. Karim, J., Somers, T., Bhattacherjee, A.: The Impact of ERP Implementation on Business Process Outcomes: A Factor-Based Study. J. Manage. Inf. Syst. 24(1), 101–134 (Jul 2007)
23. Edgar, T., Manz, D., Carroll, T.: Towards an experimental testbed facility for cyber-physical security research. Proc. of the CSIIRW '11 (2011)
24. Dijkman, R.M., Dumas, M., Ouyang, C.: Semantics and analysis of business process models in BPMN. Inf and Soft Tech 50(12), 1281–1294 (2008)
25. van der Aalst, W.M.P.: Business Process Management as the "Killer App" for Petri Nets. Software & Systems Modeling pp. 1–7 (2014)
26. van der Aalst, W.M.P.: The Application of Petri Nets to Workflow Management. Journal of Circuits, Systems and Computers 08(01), 21–66 (1998)
27. Schmutzler, J., Andersen, C.A., Wietfeld, C.: Evaluation of OCPP and IEC 61850 for Smart Charging Electric Vehicles. EV Symposium 27 (2013)
28. Wan, K., Man, K.L., Hughes, D.: Specification, analyzing challenges and approaches for cyber-physical systems (CPS). Engineering Letters 18(3) (2010)
29. Derler, P., Lee, E.a., Sangiovanni Vincentelli, A.: Modeling cyber-physical systems. Proceedings of the IEEE 100(1), 13–28 (2012)
30. Hu, H.A., Chen, S.H., Hsu, C.W., Wang, C., Wu, C.L.: Development of sustainability evaluation model for implementing product service systems. J. Environmental Science and Technology 9(2), 343–354 (2012)
31. Beuren, F.H., Gomes Ferreira, M.G., Cauchick Miguel, P.A.: Product-service systems: a literature review on integrated products and services. Journal of Cleaner Production 47, 222–231 (2013)
32. Exner, K., Sternitzke, A., Kind, S., Beckmann-Dobrev, B.: Hybrid Prototyping. In: Christoph Gengnagel, Emilia Nagy, R.S. (ed.) Rethink! Prototyping: Transdisciplinary Concepts of Prototyping, pp. 89–110. Springer (2015)
33. Johne, A., Storey, C.: New service development: a review of the literature and annotated bibliography. European Journal of Marketing 32(3/4), 184– 251 (1998)
34. Edvardsson, B., Olsson, J.: Key Concepts for New Service Development. The Service Industries Journal 16(2), 140–164 (1996)
35. Blomkvist, J.: Ways of Seeing Service: Surrogates for a Design Material. Nordes pp. 1–4 (2015)
36. Sampson, S.E.: Visualizing Service Operations. J Service Research 15(2), 182–198 (2012)
37. Blomkvist, J., Holmlid, S.: Service Prototyping According to Service Design Practitioners. Innovation pp. 1–11 (2010)
38. Bowers, M.R.: Developing new services: improving the process makes it better. Journal of Services marketing 3(1), 15–20 (1989)
39. Shostack, G.L.: How to Design a Service. E J Marketing 16(1), 49–63 (1982)
40. Becker, J., Beverungen, D., Knackstedt, R., Matzner, M., Müller, O., Pöppelbuß, J.: Bridging the Gap Between Manufacturing and Service Through IT-Based Boundary Objects. IEEE Trans Eng Man 60(3), 468–482 (2013)
41. Kostopoulos, G., Gounaris, S., Boukis, A.: Service blueprinting effectiveness: drivers of success. Managing Service Quality: an International Journal 22(6), 580–591 (2012)