sustainability

MDPI

Article

Designing Socially and Organizationally Sustainable Industry 4.0 Systems: Requirements for Modeling Approaches

Udo Kannengiesser

Institute of Business Informatics—Communications Engineering, Johannes Kepler University Linz, 4040 Linz, Austria; udo.kannengiesser@jku.at

Abstract: Industry 4.0 (I4.0) systems are often designed without sufficiently considering the needs of stakeholders and the organizational processes to be supported, leading to solutions that are socially and organizationally unsustainable. In this study, the notions of social and organizational sustainability were viewed from a micro-level perspective, referring to the ability of technology to sustain the concerns of people and work organization within the socio-technical system, as opposed to a macro-level perspective related to concerns outside the system. Through a literature review, this study shows that social and organizational sustainability is covered by principles originally proposed in agile software engineering. A set of core requirements for model-based design approaches were then derived from the agile principles, based on insights from design research and model theory. The requirements include (1) the coverage of function and behavior, (2) simplicity, (3) executability and (4) modularity. They were then used to evaluate an existing modeling approach—subject-oriented process modeling (S-BPM)—to demonstrate their applicability and usefulness.

Keywords: Industry 4.0; social sustainability; organizational sustainability; model-based design; human-centered design; agile development; process modeling; S-BPM

Citation: Kannengiesser, U. Designing Socially and Organizationally Sustainable Industry 4.0 Systems: Requirements for Modeling Approaches. *Sustainability* **2023**, *15*, 14706. https://doi.org/10.3390/su152014706

Academic Editors: Wen-Hsien Tsai and Yang (Jack) Lu

Received: 15 August 2023
Revised: 2 October 2023
Accepted: 8 October 2023
Published: 10 October 2023

1. Introduction

Today's industrial automation systems are increasingly developed as cyber-physical production systems: networks of physical assets (e.g., machines, sensors, robots, products and human operators) equipped with software and electronics to enable autonomous, flexible manufacturing and customer-oriented business models. These developments, frequently called Industry 4.0 (shorthand: I4.0) or smart manufacturing, represent a fundamental shift from traditional production systems that are based on centralized control.

I4.0 systems are highly complex, involving a multitude of interactions between heterogeneous components that often lead to non-linear system behavior [1], making the design of these systems a challenging task. According to an industry survey [2], only 14% of smart manufacturing initiatives were considered successful. Due to the high investment risk, many production companies, in particular SMEs, hesitate to undertake I4.0 projects. A recent survey found that only 34% of small and medium manufacturers in the United States have begun adopting I4.0 technologies [3].

Modeling is generally viewed as a key enabler for designing I4.0 systems, as it facilitates the understanding of complex system interactions [4,5]. Most approaches for I4.0 system design have therefore used model-based systems engineering (MBSE) methods. However, these methods are strongly technology-centric and neglect the fact that I4.0 systems need to be aligned with the business operations and human stakeholders responsible for them [6,7]. Digital technologies and the processes they afford are often perceived as too rigid to effectively support flexible work activities, which may result in employees circumventing automated procedures and systems [8–10]. In addition, common modeling notations used in MBSE, such as SysML and UML, are quite heavyweight, formal and not easy to use by domain experts [11].

These issues can be understood in terms of insufficient sustainability of I4.0 system modeling with respect to social and organizational aspects, where the social aspects include the needs and viewpoints of the stakeholders working within the I4.0 system and/or with models of the I4.0 system, and organizational aspects include the operations and processes creating value for the production company. In this study, we focused on the social and organizational aspects from an internal or micro-level perspective (i.e., related to the stakeholders and processes within the organization or socio-technical system), rather than from a macro level that considers economic, social and ecological issues outside the socio-technical system. The key to addressing micro-level sustainability for I4.0 was suggested, both by industry and academia, to be early stakeholder involvement and the use of agile design techniques [12–17]. While there have been various methods incorporating such aspects into (model-based) I4.0 systems engineering, there is no general framework for guiding or assessing the development of modeling approaches. One exception is the work by Lohmeyer et al. [11], who proposed categories of human-based and organizational aspects of systems engineering methods. Yet, that work does not include a systematically derived set of requirements specifically for I4.0 modeling approaches.

This study developed requirements for a modeling approach for I4.0 system design, built on the basic ideas of stakeholder orientation and agile methodologies. This was undertaken to enable the development of model-based I4.0 system design approaches supporting social and organizational sustainability on a micro level. This study derived the requirements from a literature review that identifies the key principles of stakeholder-oriented, agile I4.0 system design. The applicability of the requirements was shown by evaluating an existing modeling approach.

This paper is structured as follows: The notions of social and organizational sustainability on a micro level are elaborated in Section 2, including the basic assumptions of stakeholder-oriented and agile design approaches. The research methodology used in this study is described in Section 3. The results of a literature review are reported and analyzed in Section 4. They include four principles (P1−P4) that are consistent with previous accounts of agile software development. A set of core requirements (R1−R4) for modeling approaches are then derived in Section 5. Their applicability is demonstrated in Section 6, where they are used for evaluating an existing modeling approach. A discussion of the results, limitations and future work is provided in Section 7. The conclusion of this paper is given in Section 8.

2. Social and Organizational Sustainability at a Micro Level

2.1. Terminology

Sustainability has been defined in several ways depending on the specific goals and viewpoints of different scholars. In order to clarify the understanding and scope of sustainability for the purposes of this study, let us first consider the most general account of sustainability as provided in common dictionaries. This account is then elaborated according to social and organizational aspects, and viewed from two perspectives—macro level and micro level—that can be related to the academic literature on sustainability.

According to the Collins English Dictionary, the term "sustainable" is defined as "designating, of, or characterized by a practice that sustains a given condition, as economic growth or a human population, without destroying or depleting natural resources, polluting the environment, etc." The "given condition" in this definition is often further characterized using adverbs, denoting something as being, for example, "environmentally" or "socially" sustainable. In the context of technical or socio-technical systems, sustainability is understood as a relational notion, linking the system under consideration (i.e., the system that is or shall be made sustainable) to another system, such as a social, economic or ecological system. A graphical representation of this relation is shown in Figure 1 using a (mini-)concept map that defines the general meaning of "X-sustainability" of a system under consideration. Following this definition, a system was characterized in this study as socially sustainable when X is a social system (including human systems) or organizationally sustainable when X is an organization. It is important to note that the literature sometimes uses different terminology, e.g., the notion of organizational sustainability is often used to mean an organization that is sustainable with respect to environmental or societal concerns (see, for example, [18,19]).

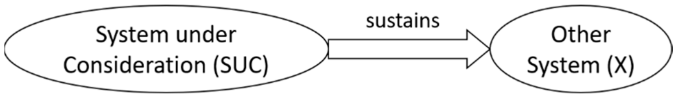
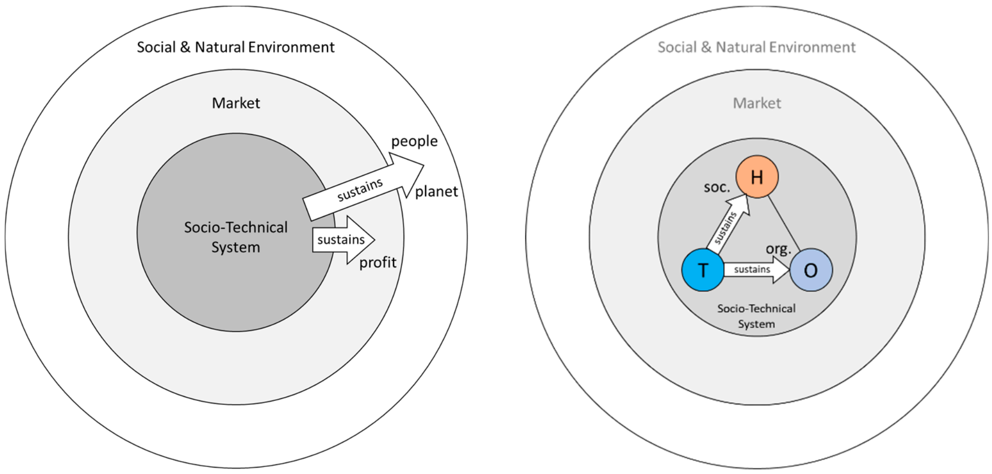


Figure 1. A system under consideration (SUC) is called "X-sustainable" if it sustains a state of another system X.

A basis for elaborating upon the notions of social and organizational sustainability used in this study and delineating them from other accounts is the human–technology–organization (HTO) model of socio-technical systems [20]. According to this model, socio-technical systems are located within a market that itself is situated within the natural and social environment. Within a socio-technical system there are three interacting subsystems: human (H), technology (T) and organization (O). The H subsystem includes the physical, biological, cognitive and cultural capabilities and concerns of the people working inside the socio-technical system. The T subsystem includes technical systems, including technical artifacts and methods, that are directly in operation or support human operations [20]. The O subsystem includes the formal and informal work organization in terms of role structures and procedures, aiming at coordinating the various organizational entities to reach a common goal. The socio-technical system can be decomposed to comprise only certain subsets of a company or enterprise, such as different departments, subsidiaries and teams. There are two useful socio-technical (sub-)systems in a manufacturing enterprise, which may be called the "design system" and the "operational system". The "design system" is a socio-technical system concerned with designing products and manufacturing systems. It comprises designers (H), design artifacts and tools (T), and design processes (O). The "operational system" is a socio-technical system concerned with using a designed manufacturing system. It comprises shopfloor workers (H), manufacturing machines (T) and manufacturing operations (O). The discussion of sustainability in this paper is primarily based on applying the HTO model to the operational system.

Similar to an approach by [21], we can distinguish different levels in the HTO model to delineate different types of sustainability depending on the system under consideration (SUC): a macro level, where the SUC is the socio-technical system, and a micro level, where the SUC is the T subsystem within the socio-technical system. On the macro level, the socio-technical system may sustain the natural environment, the social environment and/or its position on the market, as shown in Figure 2a. These types of sustainability correspond to the environmental, social and economic dimensions of Elkington's [22] triple bottom line (TBL), which is often paraphrased as planet, people and profit, respectively. More detailed aspects of macro-level sustainability were defined by the United Nations in terms of the 17 Sustainable Development Goals (SDGs).

On a micro level, the T subsystem may sustain the H and/or the O subsystem, as shown in Figure 2b. (The O subsystem may also sustain the H and/or T subsystems, and the H subsystem the T and/or O subsystems. However, we did not consider these types of sustainability in this study, as the most commonly discussed issue is how technology can sustain the other subsystems.) Specifically, we viewed the T subsystem as socially sustainable when it sustains the H subsystem, and organizationally sustainable when it sustains the O subsystem. It is important to note that on the micro level, the term "social sustainability" is understood in a different way than on the macro level: here, it is used to denote the sustainability of technology with respect to the people within the socio-technical system, rather than of the socio-technical system with respect to the people outside it.



(**a**) Macro level. (**b**) Micro level.

Figure 2. Different notions of X-sustainability located in the human–technology–organization (HTO) model of socio-technical systems: (**a**) macro level and (**b**) micro level.

2.2. Design for Social and Organizational Sustainability

In order to make systems X-sustainable, the needs of X are to be considered during system design. "Design for X" (DfX) is a general term used in engineering design research to denote approaches oriented toward taking certain design concerns into account early on in the design process [23]. Originally, DfX included approaches where "X" stands for technological concerns, such as manufacturing (DfM) or assembly (DfA). Later, sustainability has been taken into account in DfX research [24]. Design for sustainability (DfS) can be represented as shown in Figure 2 by reversing the direction of the arrows, which would then be read as "the needs of [X] are considered for the design of [the SUC]". Most of the research in DfS is based on a macro-level view, focusing on environmental and societal issues pertaining to designed products and systems. Yet, micro-level sustainability can be argued to be a pre-condition for achieving macro-level sustainability because it is not before the HTO subsystems are aligned with each other that the socio-technical system can sustain its external environment in the long run. The complexities of Industry 4.0 make this alignment more challenging than in traditional organizations. Therefore, the focus of this study was on DfS at the micro level, considering the needs of the human and organization subsystems when designing I4.0 technologies.

Creating technologies that sustain the human subsystem is one of the goals of the recent "Industry 5.0" initiative by the European Union, which aims to adapt Industry 4.0 technologies to the needs of human workers [25]. In the general area of design research, several approaches have been developed for similar purposes. A common assumption (for macro and micro levels alike) has been that human concerns can be most effectively considered by empowering stakeholders to participate in the design process [26,27]. Empowerment, generally, is conditional on "(1) access to resources and institutions, (2) strategies to mobilize them and (3) the willingness to do so" [28] (p. 512). These conditions can be supported by organizational arrangements aimed at including stakeholders in the design process. Several approaches have been developed in this context, such as human-centered design [29,30], participatory design [31], co-design [32] and design thinking [33]. In terms of the "design system" vs. "operational system" distinction presented in Section 2.1, stakeholder empowerment can be understood as blurring the boundary between the two systems by using the same H subsystem in both of them, i.e., workers becoming (co-)designers.

Approaches to sustaining the organization subsystem concentrate on aligning technologies with the changing needs of the organization that are the result of a dynamic business environment. The idea of agile design covers these approaches and is widely seen as a paradigm required for creating organizationally sustainable I4.0 systems [12–17].

Its basic values or principles, which were originally proposed in the area of software engineering [34], include:

1. Individuals and interactions over (i.e., should be valued more than) rigid procedures and tools: This concept emphasizes the importance of informal communication and self-organized work in the design process, leading to emerging forms of collaboration. Formal, fixed procedures should be reduced to a minimum, as they can reduce creativity and often represent unnecessary overhead.
2. Working systems over comprehensive documentation: This concept reflects the need for the continuous, iterative delivery of executable systems so that their usefulness can be evaluated from the perspective of the stakeholders using them. Failures to meet user expectations can thus be identified early in the design process, reducing the risk of developing wrong or ineffective solutions.
3. Customer collaboration over contract negotiation: Here, the customer can be viewed in a broad sense to refer to any adopter or stakeholder of the system being developed. Closely involving stakeholders in the design process avoids misunderstandings between system designers and system users and increases the acceptance of system designs by their users. This concept encompasses the idea of stakeholder empowerment described earlier.
4. Responding to change over following a plan: Changes during design processes occur frequently, based on new requirements, constraints or emerging opportunities. Being prepared to integrate changes in the current design is often a more successful strategy than assuming a linear (waterfall) process. It is most directly embraced by incremental approaches in which a minimum viable product (MVP) is produced and gradually extended by adding more features.

There is a lack of research on using agile and stakeholder-oriented paradigms for deriving requirements for model-based approaches for I4.0 system design. We therefore formulated the following research question (RQ): *What are the core requirements for modeling approaches to support the design of socially and organizationally sustainable I4.0 systems?*

3. Research Methodology

The methodology of this research consisted of three steps, as shown in Figure 3. In step 1, a literature review was carried out to identify the principles of agile and stakeholder-oriented I4.0 system design. It was based on the methodology for systematic literature reviews (SLRs) proposed by [35]. However, it is not claimed to be a full SLR because it was carried out by this paper's (single) author rather than by several independent reviewers as recommended by most SLR methodologies. In step 2, requirements for I4.0 modeling approaches were derived from the principles identified in step 1, based on insights from research in modeling and design. In step 3, the operationalization of the requirements was demonstrated by evaluating an existing approach commonly known as S-BPM [36]. In the remainder of this section, the three steps are described in more detail.

3.1. Step 1: Identify Principles of Stakeholder-Oriented, Agile I4.0 System Design

After formulating the research question (see Section 2.2) in the planning stage, a literature search was carried out using the Scopus database. This database was chosen because it indexes the major publishers of I4.0-related literature, including Elsevier, Emerald, IEEE, Springer and Taylor & Francis. The search terms included AND combinations of "industry 4.0" with any of the following: "stakeholder-oriented" OR "empowerment" OR "participatory" OR "co-design" OR "user-centered" OR "human-centered" OR "design thinking" OR "agile". The terms were chosen based on the micro-level DfS approaches presented in Section 2.2 and then validated based on a pilot search on Scopus. The search was limited to the titles, abstracts and keywords of journal articles, conference papers and book chapters. The types of studies considered were not limited and comprised reviews, conceptual research and empirical studies. The timeframe included any date until 30 April

2023. The search returned a list of N = 964 publications. After a screening of abstracts, the full texts were analyzed under consideration of the following criteria:

- Must be written in English;
- Must have full text available;
- Must identify concepts of agile or stakeholder-oriented I4.0 system design in terms of enablers, success factors or specific approaches;
- Agility and stakeholder orientation must refer to the process of designing the I4.0 system rather than to the results of designing (e.g., the agility of the resulting I4.0 production system), the process of production (e.g., ergonomic work tasks on the shopfloor) or the results of production (e.g., user-centered consumer products).

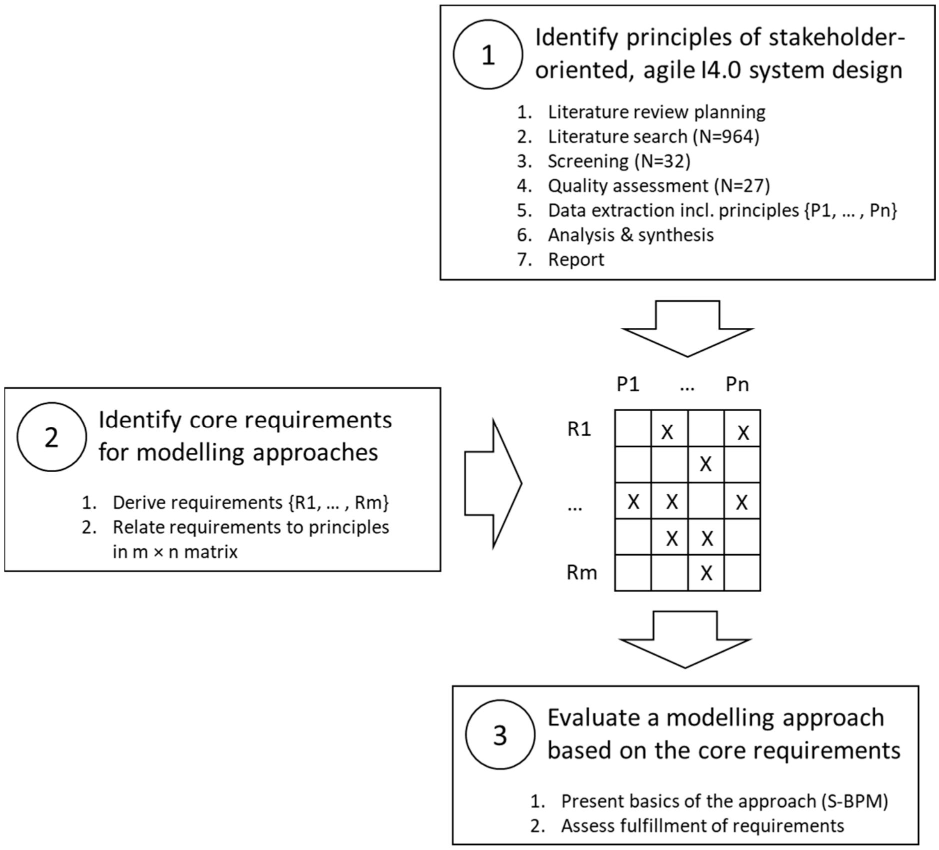


Figure 3. Methodology used in this study.

Applying these criteria reduced the number of papers to N = 32 after screening and N = 27 after a full-text analysis. The data extraction phase then consisted of collecting basic metadata and finding suitable categories for the stakeholder-oriented, agile concepts proposed or analyzed in the remaining 27 papers. The four principles stated in the agile manifesto originally proposed by [34] were found to provide a useful basis for categorization. In the analysis and synthesis stage, the extracted data was aggregated across the different papers and visualized using charts and tables. The reporting stage focused on describing the key findings in the literature review.

3.2. Step 2: Identify Requirements for Modeling Approaches

The principles found in step 1 were used as goals to be supported by modeling approaches. The required characteristics of such approaches were then derived based on research on design and modeling. This included insights into the use of conceptual models as artifacts in the process of designing. The requirements were summarized in a matrix that interrelated them with the principles, allowing for tracing them back to social and organizational sustainability goals.

3.3. Step 3: Evaluate a Modeling Approach Based on the Requirements

To demonstrate the applicability and usefulness of the requirements derived in step 2, they were used to evaluate an existing modeling approach, namely, the S-BPM methodology [36]. A presentation and subsequent analysis of the approach resulted in an evaluation table that shows whether it fulfills the requirements.

4. Principles of Agile and Stakeholder-Oriented I4.0 System Design

The distribution of the 27 relevant papers by year, publication type and research type is shown in Figure 4. All of them were published in the period of 2017–2023. All but one of the papers appeared in peer-reviewed journals and conference proceedings, which indicates high levels of quality. While most of the literature consists of conceptual contributions and SLRs, there is a reasonable share of empirical work (19%). This shows that there is sufficient grounding of the scientific claims in practice.

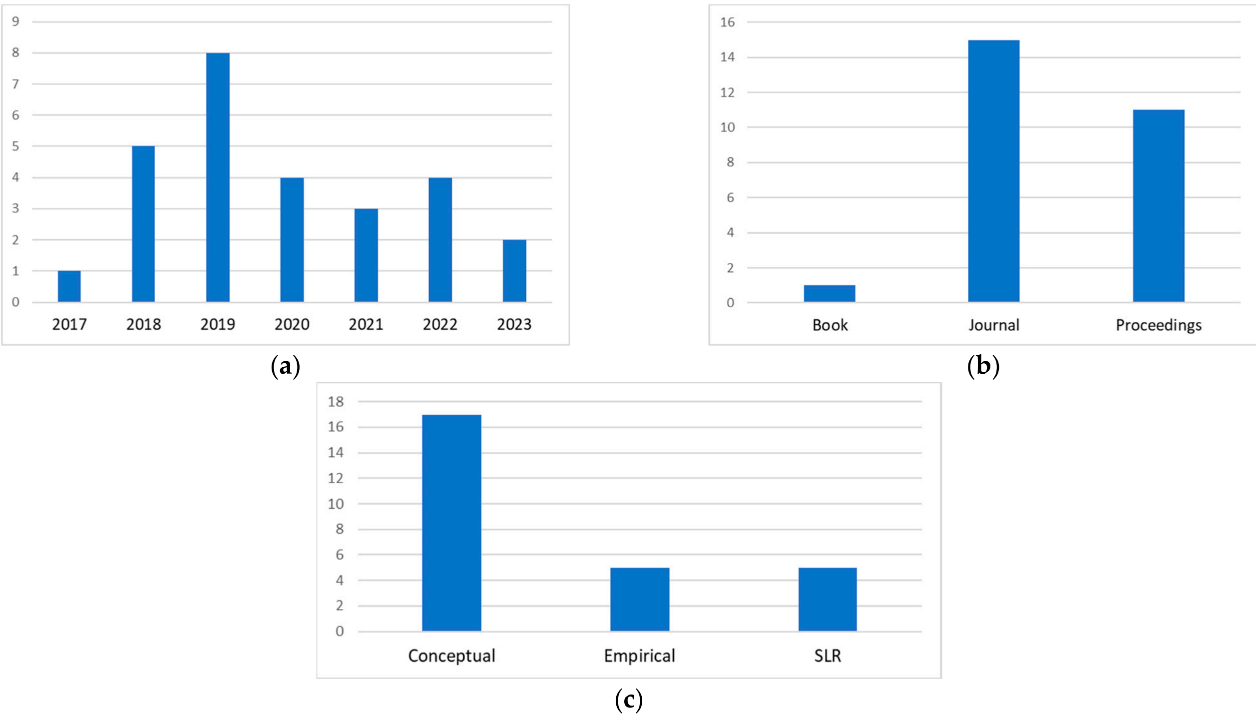


Figure 4. Distribution of relevant publications (**a**) by year, (**b**) by publication type and (**c**) by research type.

The hypothesized principles of stakeholder-oriented, agile I4.0 system design, as borrowed from the Agile Manifesto for software engineering, are all supported by the literature. The distribution of publications by principles is shown in Figure 5. Most individual publications support more than one principle. The mapping between the principles and individual publications is depicted in Table 1.

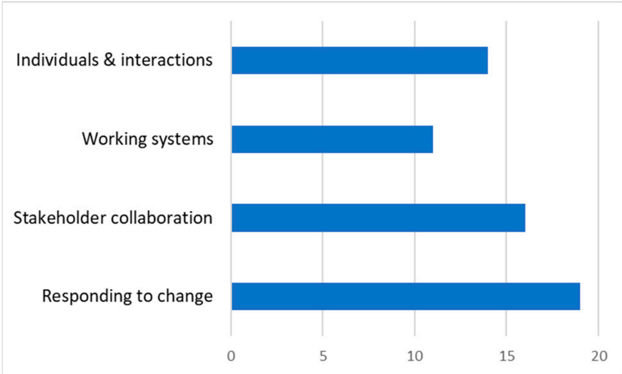


Figure 5. Distribution of relevant publications by principles of stakeholder-oriented, agile I4.0 system design.

Table 1. Literature found to be consistent with the four principles of agile I4.0 system design.

Principles	Literature
Individuals and interactions over rigid procedures and tools	Nguyen Ngoc et al. [30], de Paula et al. [37], Mule et al. [38], Martini and Bosch [39], Ericson et al. [40], Eisenträger et al. [41], Le Grand and Rébecca [42], Bauer et al. [43], Wolf et al. [44], Pfeiffer et al. [45], Sjödin et al. [46], Jussen et al. [47], Varela et al. [48], Mesa et al. [49]
Working systems over comprehensive documentation	de Paula et al. [37], Mule et al. [38], Eisenträger et al. [41], Wolf et al. [44], Jussen et al. [47], Mesa et al. [49], Christensen et al. [50], Rauch et al. [51], Kaasinen et al. [52], Niermann et al. [53], Clement et al. [54]
Stakeholder collaboration over contract negotiation	Orso et al. [17], Nguyen Ngoc et al. [30], de Paula et al. [37], Mule et al. [38], Christensen et al. [50], Ericson et al. [40], Le Grand and Rébecca [42], Bauer et al. [43], Pfeiffer et al. [45], Sjödin et al. [46], Jussen et al. [47], Mesa et al. [49], Kaasinen et al. [52], Niermann et al. [53], Vereycken et al. [55], Frysak et al. [56]
Responding to change over following a plan	Nguyen Ngoc et al. [30], de Paula et al. [37], Mule et al. [38], Martini and Bosch [39], Eisenträger et al. [41], Le Grand and Rébecca [42], Bauer et al. [43], Wolf et al. [44], Sjödin et al. [46], Jussen et al. [47], Mesa et al. [49], Christensen et al. [50], Rauch et al. [51], Clement et al. [54], Salehi [57], Salehi and Wang [58], Mrugalska and Ahmed [59], Kayabay et al. [60], Santos et al. [61]

In the remainder of this section, the ways in which the literature addresses the four principles are elaborated.

4.1. Individuals and Interactions over Rigid Procedures and Tools (Principle P1)

A common theme in the literature is the importance of having organizational structures and cultures in place that provide sufficient room for individual creativity and autonomy. These structures are characterized by flat hierarchies, decentralized decision making and unstructured ideation techniques, with few regulatory constraints and little technology-centricity [37,40,42–44]. Interactions between individuals play an equally important role: Short information paths are seen as a success factor—and, for many companies, as a challenge—for I4.0 projects [44]. More generally, the need for effective and efficient communication has been recognized [38,45,46]. Here, the focus should be on providing the involved actors with the right information (at a suitable abstraction level) at the right time, thus reducing irrelevant information flow [46]. Given the multidisciplinary and multi-departmental nature of many I4.0 projects, the issue of communication is discussed not only at an individual level but also across teams [38,39,41] and stakeholder networks [30,47]. A lack of guidance and control in coordinating different development teams has been recognized to lead to architectural technical debt (ATD) pertaining to the overall system in the sense of subsystem implementations becoming inconsistent with the globally defined architecture [39].

4.2. Working Systems over Comprehensive Documentation (Principle P2)

Working (i.e., executable) systems are implicitly assumed in the many accounts of iterative development characterized by early production and testing of prototypes. While additive manufacturing is widely accepted as a key technology for rapid prototyping in the I4.0 context, a lot of potential is also seen in virtual approaches [38,41,47,51–54]. These approaches range from the use of app mockups, simple paper storyboards [47] and process simulations [50,53,54] to 3D-realistic virtual factories [52]. Virtual models are not only more cost-efficient than physical prototypes but also allow for loosely-coupled interactions between the various disciplines involved in I4.0 design, and thus, enable parallel ways of working for different design teams [41]. Comprehensive documentation is seen as unnecessary overhead for agile I4.0 design and should be reduced to specific types of documents, namely, those describing “concept development, technical specification, drawings and interface description” [38] (p. 628).

4.3. Stakeholder Collaboration over Contract Negotiation (Principle P3)

The majority of the literature advocates for worker involvement in the early stages of I4.0 system design. The main benefits include the higher acceptance of designs by workers, better quality of design decisions [45] and high innovation potential [17]. In an industry survey [42], 97% of the respondents stated they would be more inclined to accept changes they have designed themselves. Engineers in a German automotive company were generally found to have positive attitudes toward worker involvement in I4.0 system design [45].

Precise identification of the point in the design process where stakeholders should be involved is often not provided. A few publications pinpoint the system requirements definition [38] and architecture definition [47,50,53,56] stages as the key points. Some authors propose involving stakeholders in the validation phase as a way to support designers (in system requirements and architecture definition phases) via iterative feedback loops [17,38,42,47].

While the benefits of worker involvement are recognized, there is the downside of additional costs for the project [42]. According to the survey reported in [45], missing time and “unsuitable procedures” are identified as major obstacles to involving shopfloor workers in design decisions. In addition, it was found that stakeholders are often opposed to changes, partially because of fear of job loss.

4.4. Responding to Change over Following a Plan (Principle P4)

Many authors view the complexity of I4.0 systems as a major challenge that requires effective and efficient approaches for handling changes throughout the development process. Generating system designs in small increments using frequent iterations is seen as the key enabler [37,38,46,47,49,50,61]. Feedback from downstream phases of verification and validation—sometimes even from operations and maintenance—is used for improving or extending the design in earlier phases, particularly those concerned with defining system requirements and architecture [38,39,50,61]. Modularity is seen as a basic principle for allowing such changes [46,60]. Methodologies embracing I4.0 design changes were proposed based on models from agile development, such as design thinking [49] and DevOps [47,57,58]. Others integrate agile methods in more structured, plan-driven approaches, such as the V model [38,41]. This aims to combine agility with the more systematic, rigorous nature of systems engineering that subsumes mechanical and electrical engineering, which are disciplines where the extent to which agile methods can be applied is rather limited [39]. According to [44], even within I4.0 software engineering, there is a goal conflict between the need for change and the need for stable operations. The authors of [44] propose DevOps and “Bimodal IT”—a notion originally proposed by the Gartner consulting group that denotes a combined use of routine and exploratory styles of work—as possible resolutions for this issue.

5. Core Requirements for Modeling Approaches

Having elicited the principles of agile and stakeholder-oriented I4.0 system design from the literature, a set of core requirements can be derived for model-based I4.0 design approaches. An overview of the requirements and their interrelationships with the elicited principles is shown in the form of a matrix in Table 2. They are described in the remainder of this section.

Table 2. Overview of core requirements for modeling and their mapping to the principles of stakeholder-oriented, agile I4.0 system design.

Requirements (R) for Modeling Approaches	Principles (P) of Stakeholder-Oriented, Agile I4.0 System Design			
	P1: Individuals and Interactions	P2: Working Systems	P3: Stakeholder Collaboration	P4: Responding to Change
R1: Coverage of function and behavior	X		X	
R2: Simplicity	X		X	X
R3: Executability		X	X	X
R4: Modularity	X		X	X

5.1. Coverage of Function and Behavior (Requirement R1)

One key aspect of a model—and of the modeling approach governing the construction of that model—is that it covers the concepts of the particular domain of interest [62]. The concepts used in the stakeholder-oriented design of I4.0 systems are—according to the analysis of stakeholder collaboration (principle P3) presented in Section 4.3—those occurring in the three phases of system requirements definition, architecture definition and validation. In an analysis of the INCOSE systems engineering process [63], it was found that the three phases are predominantly concerned with two ontological concepts: function and behavior. Function is defined as the purpose of a system or component. Behavior denotes the interaction of the system or component with its environment. It is only in the detailed design phases that the focus shifts from function and behavior toward the structure of the system being designed. Those phases require specific engineering knowledge that is not commonly available among stakeholders. The findings are consistent with an observation in [64] that I4.0 system design commonly follows a top-down strategy, starting with specifying business, usage and functional viewpoints before developing more technical details of the component structure.

The concepts of function and behavior are based on a black-box view of a system that corresponds to the perspective of an external observer rather than a specialist. This aligns with findings from system modeling that behavior models are preferred over structural models when little is known about the details of a system's components [65]. Behavior models were found to increase the human understanding of complex systems and represent a key concern of stakeholders [66]. Function models, which abstract from specific structures and behaviors, can be accessed and used with a reduced cognitive load [67]. They afford the perspective of an "overall picture", matching the general preference of people for understanding the whole system before attending to its parts [68]. Therefore, function and behavior models can be useful for coordination across different disciplines, thus enhancing the interaction between individuals within and across teams (P1).

5.2. Simplicity (Requirement R2)

The notion of simplicity can be defined generally as the set of qualities of a modeling approach that contribute to a low level of effort required to produce and interpret models. These qualities encompass various syntactic and semantic properties and their relationships.

One of the most important factors for the simplicity of a modeling approach is the number of semantic constructs it contains [69]. Fewer constructs reduce the effort of learning and using the approach, which is especially critical for most I4.0 stakeholders who are not trained in system modeling. Other factors include the suitability of the approach to be used with simple, intuitive tools—tools that do not require much effort in learning and usage by untrained stakeholders. Anecdotal evidence suggests that there is a strong correlation between the complexity of the approach and the complexity of the modeling tools required [70]. The simpler the constructs and tools of the modeling approach, the easier and more ad hoc becomes the interaction between the individuals involved (P1) and their joint construction of a system model. Simplicity also implies abstraction because it is the result of omitting information. Simple, abstract models provide common ontologies that facilitate communication between stakeholders that have different backgrounds [71] (P3). Finally, simplicity can be seen as enhancing responsiveness to change, as every change involves constructing a new, modified model. The simpler the modeling approach, the more responsive the model is to change (P4).

5.3. Executability (Requirement R3)

Executability is a feature of a modeling approach that allows for transforming models into working systems (P2), which can be used either directly for the implementation of the target system or for simulations executed by a "proxy" system. Executability is based on the availability of formally defined execution semantics for the model. Having an executable model has the advantage that no manual effort is required for the transformation, reducing the cost of changes to an existing model (P4) and the risk of misinterpretations of the model by human implementers. It also allows for rapid prototyping in short, iterative cycles of design and testing. The target systems of model transformation in I4.0 need to include execution technologies for cyber-physical production systems, manufacturing execution systems, programmable logic controllers (PLCs) and other control systems on the shop floor. Existing standards for some of these technologies include IEC 61131-3 [72] and IEC 61499 [73].

In agile software engineering, the motivation for producing executable models was found to be higher than for non-executable ones because of their direct impact on implementation [74]. When the models represent services, "service walkthroughs" can be executed in real or simulated environments that can lead to better comprehension and acceptance of the modeled services [75]. A variety of tangible and immersive technologies may be used to enhance this effect [76]. In conclusion, executability also contributes to stakeholder involvement (P3).

5.4. Modularity (Requirement R4)

In the domain of modeling, modularity is generally understood as the ability to organize a model into a set of interconnected subsystems (also called modules) with reduced dependencies [77,78]. It increases the changeability (P4), as the effects of changes can be limited to individual modules, without necessarily propagating to other parts of a system [79]. Modularity also enhances the reuse of modules across different models, thus facilitating the creation of new models and the modification of existing ones [80]. Incremental design strategies that begin with developing a set of core features to produce a minimum viable product (MVP), which is then gradually augmented and modified, are particularly dependent on modular structures. One additional benefit is that modularity facilitates "divide-and-conquer" approaches: A complex system design task can be broken down into a set of loosely coupled subtasks (i.e., modules), each of which can be assigned to different stakeholders [81]. This enables effective stakeholder involvement, as the modules can be defined to match the stakeholders' individual areas of responsibility and expertise (P1). Modularity makes the development, deployment and evolution of systems more flexible and scalable [82]. Finally, models that explicitly represent the interfaces between modules (e.g., architecture models) provide effective support for the coordination of the stakeholders (P3) assigned to the respective modules [74]. This is because the models encapsulate, and thus hide, details that are internal to a module and not relevant for the interplay with other modules. Internal details can be added by the respective stakeholder in their own time, independently of their peers' schedules. It is only in the case of interfaces needing to be (re-)defined that the stakeholders need to coordinate [83].

6. Evaluating an Existing Modeling Approach

We can demonstrate the applicability and usefulness of the requirements by using them for the evaluation of existing modeling approaches. In this section, a modeling approach commonly known as subject-oriented business process management (S-BPM) [36] is evaluated based on the requirements. The S-BPM approach was originally developed for modeling and executing business information systems but has increasingly been applied to cyber-physical system design in production and other domains [84–87]. In this approach, systems are conceived of as interactions between functional entities called "subjects". Subjects coordinate their individual behaviors by exchanging messages with one another. This section commences with an introduction to the basic concepts of S-BPM using a very simple example of an order management process to facilitate understanding by most readers. A more complex example of a robot-based package handling process is shown later, demonstrating the applicability of S-BPM in the domain of Industry 4.0. Finally, this section evaluates the S-BPM approach with respect to the requirements derived in Section 5. More details of the S-BPM methodology and notation can be found in [36].

S-BPM modeling uses two types of diagrams: subject interaction diagrams (SIDs), which describe the subjects and the exchange of messages between them, and subject behavior diagrams (SBDs), which specify the behavior of individual subjects. The SID of an order management process for spare parts is shown in Figure 6. Arrows in the SID represent messages exchanged between subjects. A message consists of a piece of information that can be a simple signal or a complex data object. A message can also be used for denoting physical objects exchanged between subjects, such as the *spare parts* transferred from *Production* to *Shipment*.

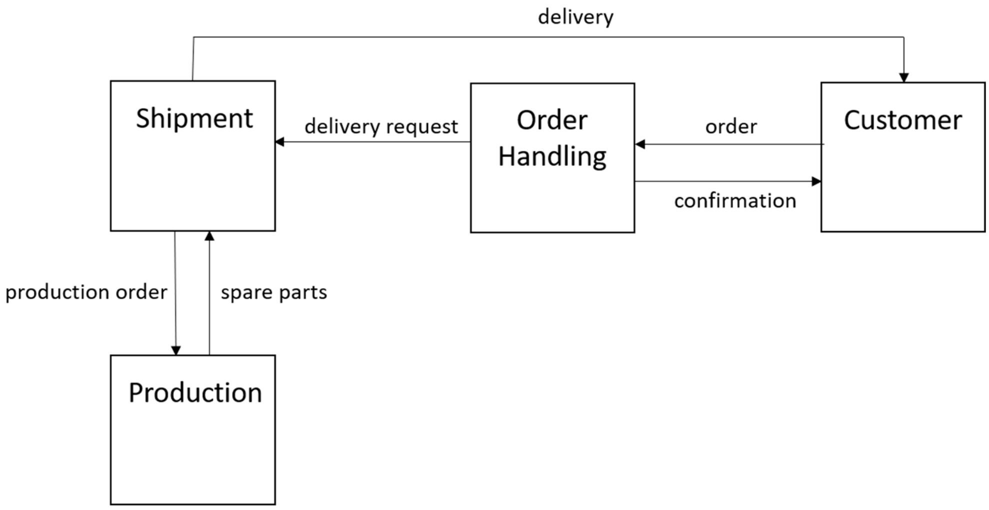


Figure 6. Subject interaction diagram (SID) of an order management process for spare parts.

The detailed behavior of every subject is specified using an individual subject behavior diagram (SBD). An example of an SBD is shown in Figure 7, which describes the behavior of the *Shipment* subject. It is a directed graph that connects three types of nodes: *Do states* (representing actions), *Receive states* (representing receipt of a message from another subject) and *Send states* (representing dispatch of a message to another subject). The arrows represent state transitions that become active once the preceding state has been executed. Conditions may be added to transitions to enable XOR branching. Parallel (AND) branches are not allowed with a single SBD. They must be represented using separate subjects for every individual behavior.

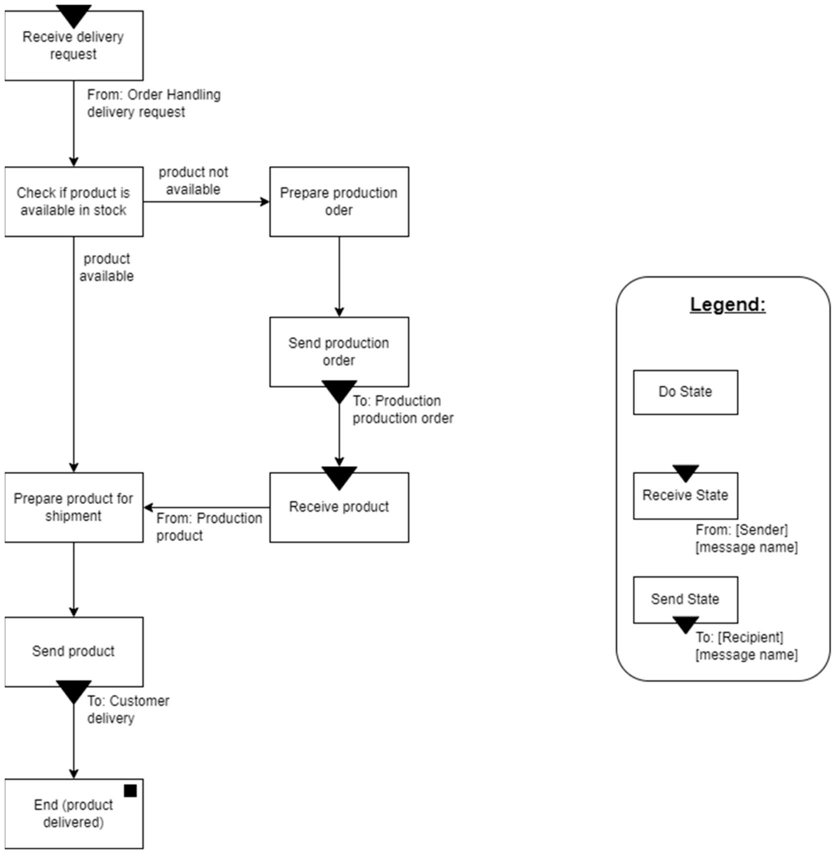


Figure 7. Subject behavior diagram (SBD) of the "Shipment" subject.

The constructs of S-BPM are quite generic and can be applied to process modeling in any domain. An example from Industry 4.0 is represented as an SID in Figure 8. This model was produced using a commercial S-BPM modeling tool. The blue boxes correspond to subjects, and the white boxes on arrows correspond to messages. The SID represents an automated package handling process, where packages arrive that need to be placed in smart transport boxes equipped with sensors that can monitor the state of the packaged items (e.g., pharmaceutical products that may be sensitive to temperature). The process is triggered by a light barrier upon detecting the arrival of a package at a workstation. A scanner is used by a robot to read the package label and identify which sensors are required for the transport box. The robot then uses its arm units to collect the sensors from a shelf, mount them inside the box and place the package in that box.

The explanation of this process is not further elaborated here, as the evaluation to be carried out in this section is independent of the particular example used. For more examples of using S-BPM in the manufacturing domain, readers are referred to [85].

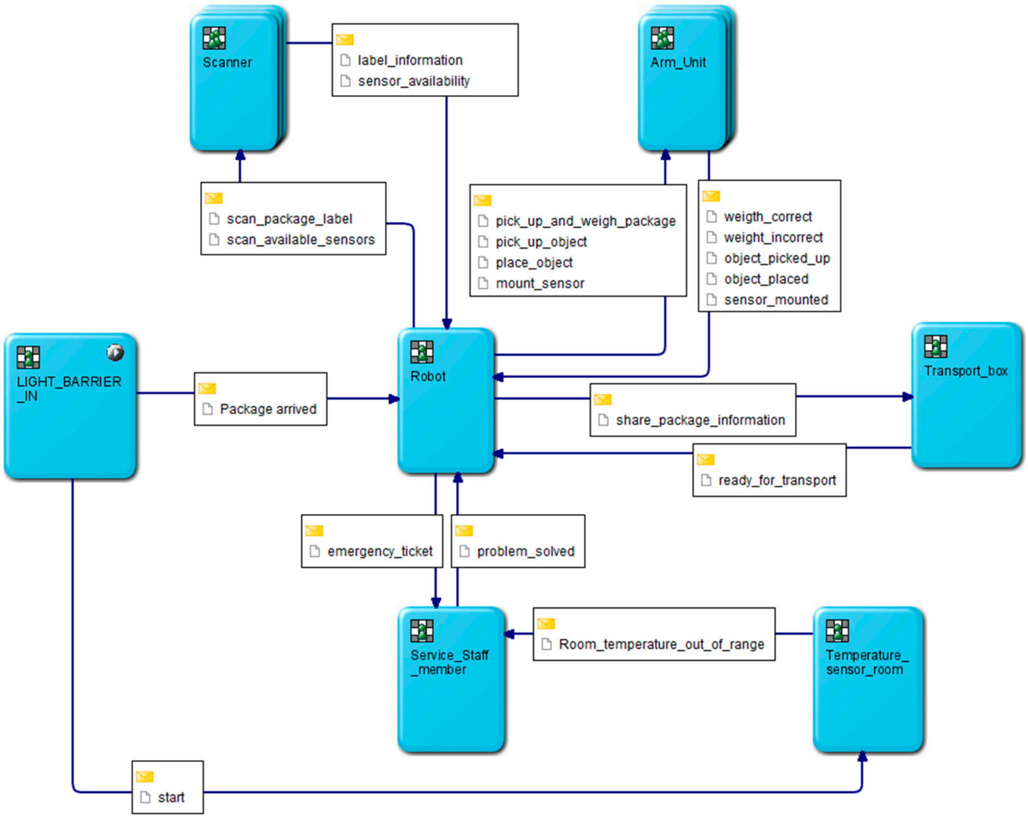


Figure 8. Subject interaction diagram (SID) of a robot-based package handling process.

6.1. Coverage of Function and Behavior

Subjects are ultimately executed by human or computational actors. However, the notion of a subject is defined as a process-centered functionality. By abstracting from the specific actors, subjects allow for greater flexibility, as the same functions can be performed by different actors. For example, in Figure 6, the *Shipment* and *Production* subjects may be executed by different service providers and factories. Subject interaction diagrams (SIDs) can be seen as architectural models of a system that are similarly based on the composition of functional entities [88]. The use of messages for representing material flows and information flows is consistent with functional models in engineering design [89]. Sequences of the subjects' actions and interactions compose their behavior. Therefore, S-BPM covers both function and behavior, which allows for modeling how a system is to behave without necessarily having to know the structural "mechanics" of its components. As a result, common ground between stakeholders, as well as between experts from different disciplines, can be reached more effectively and efficiently.

6.2. Simplicity

One of the most characteristic features of modeling in S-BPM is the reduced set of notational elements and model types compared with similar, process-oriented approaches, such as BPMN, which has over 160 elements. Only three basic constructs (*Do states*, *Receive states* and *Send states*) are needed for SBD modeling, and only two (*Subject* and *Message*) for SID modeling. This facilitates learning and correctly applying the notation, especially for stakeholders not trained in modeling. The notational simplicity comes with the possibility to use simple tools for fast, ad hoc ways of modeling. Such tools range from pen and paper, cards, whiteboards and office software (e.g., MS Visio version 2023 and Excel 2019) to high-tech, tangible tabletop interfaces [90]. There is anecdotal and empirical evidence for the ease and speed with which novices can learn S-BPM modeling. In a study by Fleischmann [91], factory workers were instructed in the approach for only 20–30 min before they were able to produce correct S-BPM models. In experiments reported in [92], novice modelers producing S-BPM diagrams significantly outperformed those producing BPMN diagrams in terms of modeling time and model quality. In three digitalization projects in the manufacturing industry [93], workers were able to produce S-BPM models of their own workplaces after a few minutes of introduction to S-BPM and with modeling support provided by a facilitator.

6.3. Executability

S-BPM models provide visual diagrams consistent with the formalism of abstract state machines (ASMs) [94], enabling their instant transformation into executable workflows. A range of open-source and commercial systems are able to execute these workflows in productive or simulation environments (www.i2pm.net/category/tools, accessed on 7 October 2023). Müller [95] showed that S-BPM models can be transformed into IEC 61131-3 [72] sequential function charts (SFCs) that can be executed by PLCs, and thus, be used for manufacturing control. On the other hand, SFCs are seen as insufficient for providing the decentralized control capabilities needed for Industry 4.0 [96]. Therefore, the executability requirement is only partially fulfilled by S-BPM in the context of I4.0 systems.

The executability of the models in (non-real-time) IT environments has been demonstrated to contribute to stakeholder engagement and their continued use throughout the system design process as a practical tool rather than just for documentation [93]. S-BPM models readily provide test scenarios for unit testing [97] and acceptance testing [93].

6.4. Modularity

SIDs are modular based on the encapsulation of behavioral details of subjects in the respective SBDs. Interfaces are established by the message exchanges defined between them. As long as these interfaces remain the same, the internal behavior of a subject can be changed independently of the behaviors of other subjects. This has the benefit that stakeholders can model individually and in parallel with other stakeholders, leading to overall time savings of about 40% compared with the non-modular BPMN approach [92]. In the order management example, the behavior of the *Shipment* subject can be modeled completely independently of the other subjects as long as the agreed message exchanges are incorporated as *Send* and *Receive states* in the corresponding SBD. A modeling method based on S-BPM that explicitly comprises alternating phases of individual modeling (concentrating on SBDs) and collaborative alignment (concentrating on the SID) was developed by [98] and applied to I4.0 system modeling by [99].

With S-BPM, I4.0 systems are modeled from the perspective of a single process type. For many single-purpose systems, such as a simple conveyor belt, such a view is sufficient. Other systems, such as autonomous guided vehicles (AGVs) or robotic systems may be involved in more than one process type. For these multi-process systems, S-BPM models can be seen as analogous to minimum viable products (MVPs) that specify only one “feature” (process). Adding S-BPM models of other “features” or processes then leads to a complete specification of the system.

6.5. Evaluation Summary

The results of the evaluation are summarized in Table 3. It can be seen that S-BPM fulfills almost all of the requirements of a modeling approach to support stakeholder-oriented, agile I4.0 system design. It is only the missing readiness to execute S-BPM models on decentralized I4.0 control infrastructures that reduces the capacity of S-BPM to be used as an effective modeling approach for socially and organizationally sustainable I4.0 systems.

Table 3. Summary of the evaluation of S-BPM. Requirements may be fulfilled (+), partially fulfilled (+/−) or not fulfilled (−).

Requirements	Fulfillment
R1: Coverage of function and behavior	+
R2: Simplicity	+
R3: Executability	+/−
R4: Modularity	+

7. Discussion

The evaluation of the S-BPM approach demonstrated the applicability of the requirements identified in this study for purposes of assessing, selecting and potentially extending modeling approaches for I4.0 system design. The requirements can therefore be seen as a step toward operationalizing the development of modeling approaches for (micro-level) sustainable I4.0 system design. The focus on modeling approaches may complement current efforts in developing MBSE methodologies with increased individual and organizational acceptance [100].

When shifting the focus from the "operational system" to the "design system" (see Section 2.1), an I4.0 modeling approach can be viewed as (part of) the technology (T) subsystem in the MTO model, as shown in Figure 9. The requirements (R1−R4) support the sustainability of the approach with respect to the concerns of design stakeholders (H) and design processes (O). In the figure, the positioning of R1−R4 with respect to the two arrows indicates their emphasis on either social or organizational sustainability in the design system. The coverage of function and behavior (R1) (*what* is represented) and simplicity (R2) (*how* it is represented) strongly facilitate human comprehension of I4.0 system designs, while their impact on the design process is rather low. They are thus located near the arrow representing social sustainability and further away from the one representing organizational sustainability. In turn, executability (R3) and modularity (R4) strongly support rapid iterations and incremental change, which are most directly connected to design process aspects. They are therefore located closer to the arrow representing organizational sustainability.

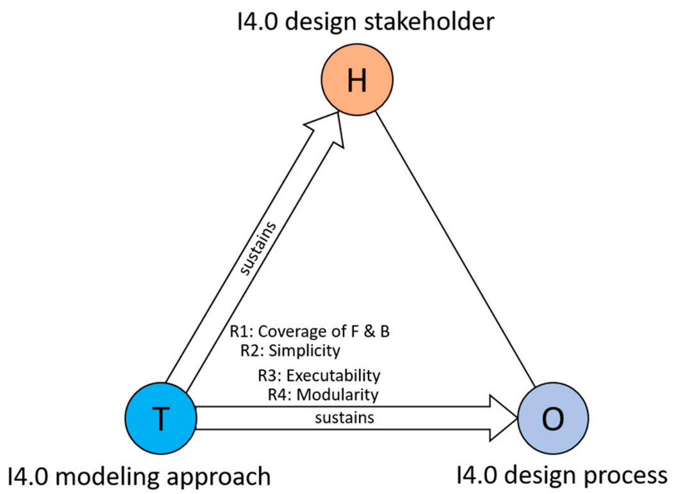


Figure 9. The four core requirements (R1−4) for I4.0 modeling approaches located on a spectrum between social and organizational sustainability within the design system.

The "design system" view depicted in Figure 9 is independent of the macro vs. micro level distinction made for the "operational system" view. It may be possible that the design stakeholders include external actors, e.g., citizens, governments and NGOs, depending on the scope of the I4.0 system design and its relevance to the general public. They may also include networks of organizations with common goals pertaining to economic, ecological or other macro-level improvements [101]. The four requirements are likely to remain the same for such an extended focus of sustainability from the micro to macro levels. Therefore, existing modeling approaches for involving external stakeholders [102] may also be examined with respect to the requirements identified in this study.

This study has assessed the S-BPM approach based on conceptual analyses of its notational constructs and on available evidence from empirical studies. Future research may focus on similar evaluations using other modeling approaches, such as SysML, UML and BPMN, allowing for comparative statements based on their respective strengths and weaknesses. To date, such a comparison has been done only between S-BPM and BPMN [92]. Identified weaknesses may drive further research investigating how the various approaches can be improved to better align I4.0 design with stakeholder-oriented, agile principles. In the case of S-BPM, its lack of executability in decentralized manufacturing environments is planned to be addressed by creating mappings between S-BPM models and I4.0 control standards, such as IEC 61499 [96].

There are a few limitations that also represent directions for future research. One such limitation is the coarse-grained level of granularity with which the requirements were described. Take the requirement of "simplicity" of a modeling approach: It depends not only on the number of semantic elements in the approach but also on the number of possible compositions of these elements as defined in a notational grammar. The symbolic appearance of elements (e.g., shape and color) also plays a role in the ease with which modelers can interpret and construct models [69,70]. In addition, further research may specify standardized metrics and test scenarios in order to quantitatively measure and compare the performances of different modeling approaches. This involves defining uniform modeling tasks to be performed in controlled settings, similar to the comparison of modeling with S-BPM vs. BPMN in the experiments conducted by [92].

Another issue is that the requirements related to social sustainability are based mostly on the cognitive, information-processing needs of stakeholders. However, the H subsystem also comprises other aspects, such as physical, biological and cultural characteristics. For example, motivation is a major prerequisite for empowerment [28], which for a variety of reasons is not always available among factory workers [85]. In addition, stakeholders may not always have sufficient knowledge to provide useful input into an I4.0 system design. Their expertise is often restricted to their own workplace and existing ways of working, lacking the essential competences needed for imagining novel manufacturing scenarios beyond Industry 3.0. Such competences include systems thinking, future-open thinking and strategic thinking [103]. It remains to be explored to what extent a modeling approach is able to support these modes of thinking.

This study focused only on core requirements—those directly derived from the micro-level sustainability perspective. Future work should also investigate whether there are additional requirements. For example, there is a need to integrate a stakeholder-oriented modeling approach for the early design phases—mostly concerned with the functions and behaviors of the I4.0 system—with the ones used by mechanical and electrical engineers, software experts, etc., in the later phases that are mainly concerned with system structure. It fits with recent research efforts to augment digital twin models that today are predominantly structurally oriented with information regarding behavior [86,104]. Mappings between different modeling approaches may, therefore, be required to allow for model transformations across design phases. Such research fits with recent extensions of the VDI 2206 guideline for mechatronic system development to include seamless modeling throughout the entire system lifecycle [105].

8. Conclusions

The notion of micro-level sustainability is the result of adopting a socio-technical perspective of I4.0 systems that was found to increase the adoption of I4.0 technologies by production companies [7], supporting the United Nation's Sustainable Development Goal 9 "Industry, Innovation and Infrastructure". This study contributes to this effect by identifying core requirements for modeling approaches supporting the socio-technical perspective of I4.0 system design. They include (1) coverage of function and behavior, (2) simplicity, (3) executability and (4) modularity. The requirements are grounded in the basic principles of stakeholder-oriented, agile design in an I4.0 context, as extracted and elaborated using systematic literature review techniques. The applicability and usefulness of the requirements were shown by evaluating an existing approach for modeling I4.0 systems.

The scope for future work is proposed to include two broad areas: First, the requirements should be used for assessing and enhancing individual I4.0 modeling approaches. Here, work is already underway to improve the executability of S-BPM in I4.0 environments by mapping it to the IEC 61499 control standard [73]. Second, the requirements should be extended by considering more aspects of social and organizational sustainability and further operationalized by developing more detailed metrics and test scenarios.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Horváth, I. What the design theory of social-cyber-physical systems must describe, explain and predict? In *An Anthology of Theories and Models of Design*; Chakrabarti, A., Blessing, L.T.M., Eds.; Springer: London, UK, 2014; pp. 99–120. [CrossRef]
2. Capgemini. *Smart Factories @ Scale: Seizing the Trillion-Dollar Prize through Efficiency by Design and Closed-Loop Operations*; Capgemini: Paris, France, 2019; Available online: https://www.capgemini.com/wp-content/uploads/2019/11/Report-%E2%80%93-Smart-Factories.pdf (accessed on 7 October 2023).
3. Pixley, J.E.; Kunrath, S.; Paz, G.; Li, B.; Donovan, R.; Li, G.P. *Enabling Small and Medium Manufacturers to Adopt Smart Manufacturing*; The American Council for an Energy-Efficient Economy (ACEEE): Washington, DC, USA, 2021.
4. Wortmann, A.; Combemale, B.; Barais, O. A systematic mapping study on modeling for industry 4.0. In Proceedings of the 2017 ACM/IEEE 20th International Conference on Model Driven Engineering Languages and Systems (MODELS), Austin, TX, USA, 17–22 September 2017; pp. 281–291. [CrossRef]
5. Henderson, K.; Salado, A. Value and benefits of model-based systems engineering (MBSE): Evidence from the literature. *Syst. Eng.* **2021**, *24*, 51–66. [CrossRef]
6. Xu, X.; Lu, Y.; Vogel-Heuser, B.; Wang, L. Industry 4.0 and Industry 5.0—Inception, conception and perception. *J. Manuf. Syst.* **2021**, *61*, 530–535. [CrossRef]
7. Marcon, E.; Soliman, M.; Gerstlberger, W.; Frank, A.G. Sociotechnical factors and Industry 4.0: An integrative perspective for the adoption of smart manufacturing technologies. *J. Manuf. Technol. Manag.* **2022**, *33*, 259–286. [CrossRef]
8. Kadir, B.A.; Broberg, O. Human well-being and system performance in the transition to industry 4.0. *Int. J. Ind. Econ.* **2020**, *76*, 102936. [CrossRef]
9. Lammi, I.J. Automating to control: The unexpected consequences of modern automated work delivery in practice. *Organization* **2021**, *28*, 115–131. [CrossRef]
10. Larsson, C.E.; Andersen, B.; Martinsen, K. Workarounds in application and use of manufacturing software as enablers to organizational change. *Procedia CIRP* **2021**, *104*, 1954–1959. [CrossRef]
11. Lohmeyer, Q.; Albers, A.; Radimersky, A.; Breitschuh, J. Individual and organizational acceptance of systems engineering methods—Survey and recommendations. In Proceedings of the TMCE 2014, Budapest, Hungary, 19–23 May 2014; Horváth, I., Rusák, Z., Eds.; pp. 1531–1540.
12. Abramovici, M. (Ed.) *Engineering Smarter Produkte und Services (Plattform Industrie 4.0 Studie)*; Acatech—Deutsche Akademie der Technikwissenschaften: München, Germany, 2018.
13. VDMA. *Guideline Industrie 4.0: Guiding Principles for the Implementation of Industrie 4.0 in Small and Medium Sized Businesses*; Verband Deutscher Maschinen- und Anlagenbau: Frankfurt, Germany, 2016.
14. PwC. *Digital Champions: How Industry Leaders Build Integrated Operations Ecosystems to Deliver End-to-End Customer Solutions*; Global Digital Operations Study 2018; PricewaterhouseCoopers: London, UK, 2018.
15. VDI. *Testing of Networked Systems for Industrie 4.0*; Verein Deutscher Ingenieure: Düsseldorf, Germany, 2018.
16. Dumitrescu, R.; Albers, A.; Riedel, O.; Stark, R.; Gausemeier, J. (Eds.) *Advanced Systems Engineering: Value Creation in Transition*; Fraunhofer IEM: Paderborn, Germany, 2021.
17. Orso, V.; Ziviani, R.; Bacchiega, G.; Bondani, G.; Spagnolli, A.; Gamberini, L. Employee-centric innovation: Integrating participatory design and video-analysis to foster the transition to Industry 5.0. *Comput. Ind. Eng.* **2022**, *173*, 108661. [CrossRef]

18. Braccini, A.M.; Margherita, E.G. Exploring organizational sustainability of Industry 4.0 under the Triple Bottom Line: The case of a manufacturing company. *Sustainability* **2019**, *11*, 36. [CrossRef]
19. Mohrman, S.A.; Worley, C.G. The organizational sustainability journey: Introduction to the special issue. *Organ. Dyn.* **2010**, *39*, 289–294. [CrossRef]
20. Karltun, A.; Karltun, J.; Berglund, M.; Eklund, J. HTO—A complementary ergonomics approach. *Appl. Ergon.* **2017**, *59*, 182–190. [CrossRef]
21. Zinke-Wehlmann, C.; Friedrich, J.; Kirschenbaum, A.; Wölke, M.; Brückner, A. Conceptualizing sustainable artificial intelligence development. In *Collaborative Networks in Digitalization and Society 5.0, Proceedings of the PRO-VE 2022, Lisbon, Portugal, 19–21 September 2022*; Camarinha-Matos, L.M., Ortiz, A., Boucher, X., Osório, A.L., Eds.; IFIP Advances in Information and Communication Technology; Springer: Cham, Switzerland, 2022; Volume 662, pp. 545–554. [CrossRef]
22. Elkington, J. Accounting for the triple bottom line. *Meas. Bus. Excell.* **1998**, *2*, 18–22. [CrossRef]
23. Kuo, T.-C.; Huang, S.H.; Zhang, H.-C. Design for manufacture and design for 'X': Concepts, applications, and perspectives. *Comput. Ind. Eng.* **2001**, *41*, 241–260. [CrossRef]
24. Ceschin, F.; Gaziulusoy, I. Evolution of design for sustainability: From product design to design for system innovations and transitions. *Des. Stud.* **2016**, *47*, 118–163. [CrossRef]
25. European Commission. *Enabling Technologies for Industry 5.0*; European Commission, Directorate-General for Research and Innovation: Brussels, Belgium, 2020.
26. Corsini, L.; Moultrie, J. Design for social sustainability: Using digital fabrication in the humanitarian and development sector. *Sustainability* **2019**, *11*, 3562. [CrossRef]
27. Zhang, M. Design empowerment: Participatory design towards social sustainability. In *Cross-Cultural Design. Interaction Design Across Cultures, Proceedings of the HCII 2022, Virtual Event, 26 June–1 July 2022*; Rau, P.L.P., Ed.; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2022; Volume 13311, pp. 274–287. [CrossRef]
28. Avelino, F. Power in sustainability transitions: Analysing power and (dis)empowerment in transformative change towards sustainability. *Environ. Policy Gov.* **2017**, *27*, 505–520. [CrossRef]
29. Boy, G.A. *Orchestrating Human-Centered Design*; Springer: London, UK, 2013.
30. Nguyen Ngoc, H.; Lasa, G.; Iriarte, I. Human-centred design in industry 4.0: Case study review and opportunities for future research. *J. Intell. Manuf.* **2022**, *33*, 35–76. [CrossRef] [PubMed]
31. Spinuzzi, C. The methodology of participatory design. *Tech. Commun.* **2005**, *52*, 163–174.
32. Steen, M.; Manschot, M.A.J.; de Koning, N. Benefits of co-design in service design projects. *Int. J. Des.* **2011**, *5*, 53–60.
33. Brown, T. *Change by Design: How Design Thinking Transforms Organizations and Inspires Innovation*; Harper-Collins: New York, NY, USA, 2009.
34. Fowler, M.; Highsmith, J. The agile manifesto. *Softw. Dev.* **2001**, *9*, 28–35.
35. Xiao, Y.; Watson, M. Guidance on conducting a systematic literature review. *J. Plan. Educ. Res.* **2019**, *39*, 93–112. [CrossRef]
36. Fleischmann, A.; Schmidt, W.; Stary, C.; Obermeier, S.; Börger, E. *Subject-Oriented Business Process Management*; Springer: Berlin/Heidelberg, Germany, 2012. [CrossRef]
37. De Paula, D.; Marx, C.; Wolf, E.; Dremel, C.; Cormican, K.; Uebernickel, F. A managerial mental model to drive innovation in the context of digital transformation. *Ind. Innov.* **2023**, *30*, 42–66. [CrossRef]
38. Mule, S.; Plateaux, R.; Hehenberger, P.; Penas, O.; Patalano, S.; Vitolo, F. A new agile hybridization approach and a set of related guidelines for mechatronic product development. In *Product Lifecycle Management Enabling Smart X, Proceedings of the PLM 2020, Rapperswil, Switzerland, 5–8 July 2020*; Nyffenegger, F., Ríos, J., Rivest, L., Bouras, A., Eds.; IFIP Advances in Information and Communication Technology; Springer: Cham, Switzerland, 2020; Volume 594, pp. 618–633. [CrossRef]
39. Martini, A.; Bosch, J. Architectural technical debt in embedded systems. In *Systems Engineering in the Fourth Industrial Revolution: Big Data, Novel Technologies, and Modern Systems Engineering*; Kenett, R.S., Swarz, R.S., Zonnenshain, A., Eds.; John Wiley & Sons: Hoboken, NJ, USA, 2020; pp. 77–103. [CrossRef]
40. Ericson, A.; Lugnet, J.; Solvang, W.D.; Kaartinen, H.; Wenngren, J. Challenges of Industry 4.0 in SME businesses. In Proceedings of the 2020 3rd International Symposium on Small-Scale Intelligent Manufacturing Systems (SIMS), Gjovik, Norway, 10–12 June 2020; pp. 1–6. [CrossRef]
41. Eisenträger, M.; Adler, S.; Kennel, M.; Möser, S. Changeability in engineering. In Proceedings of the 2018 IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC), Stuttgart, Germany, 17–20 June 2018; pp. 1–8. [CrossRef]
42. Le Grand, T.; Rébecca, D. COOC: An agile change management method. In Proceedings of the 2019 IEEE 21st Conference on Business Informatics (CBI), Moscow, Russia, 15–17 July 2019; pp. 28–37. [CrossRef]
43. Bauer, W.; Schuler, S.; Hornung, T.; Decker, J. Development of a procedure model for human-centered Industry 4.0 projects. *Procedia Manuf.* **2019**, *39*, 877–885. [CrossRef]
44. Wolf, M.; Semm, A.; Erfurth, C. Digital transformation in companies—Challenges and success factors. In *Innovations for Community Services, Proceedings of the I4CS 2018, Žilina, Slovakia, 18–20 June 2018*; Hodoň, M., Eichler, G., Erfurth, C., Fahrnberger, G., Eds.; Communications in Computer and Information Science; Springer: Cham, Switzerland, 2018; Volume 863, pp. 178–193. [CrossRef]
45. Pfeiffer, S.; Lee, H.; Held, M. Doing Industry 4.0—Participatory design on the shop floor in the view of engineering employees. *Cuad. De Relac. Laborales* **2019**, *37*, 293–311. [CrossRef]

46. Sjödin, D.R.; Parida, V.; Leksell, M.; Petrovic, A. Smart factory implementation and process innovation. *Res. Technol. Manag.* **2018**, *61*, 22–31. [CrossRef]
47. Jussen, P.; Kuntz, J.; Senderek, R.; Moser, B. Smart service engineering. *Procedia CIRP* **2019**, *83*, 384–388. [CrossRef]
48. Varela, L.; Putnik, G.; Romero, F. The concept of collaborative engineering: A systematic literature review. *Prod. Manuf. Res.* **2022**, *10*, 784–839. [CrossRef]
49. Mesa, D.; Renda, G.; Gorkin, R., III; Kuys, B.; Cook, S.M. Implementing a Design Thinking Approach to De-Risk the Digitalisation of Manufacturing SMEs. *Sustainability* **2022**, *14*, 14358. [CrossRef]
50. Christensen, H.B.; Jepsen, S.C.; Worm, T. Agile architecting of distributed systems for flexible Industry 4.0. *Ann. Comput. Sci. Inf. Syst.* **2021**, *25*, 533–536. [CrossRef]
51. Rauch, E.; Linder, C.; Dallasega, P. Anthropocentric perspective of production before and within Industry 4.0. *Comput. Ind. Eng.* **2020**, *139*, 105644. [CrossRef]
52. Kaasinen, E.; Aromaa, S.; Heikkilä, P.; Liinasuo, M. Empowering and engaging solutions for Operator 4.0—Acceptance and foreseen impacts by factory workers. In *Advances in Production Management Systems. Production Management for the Factory of the Future, Proceedings of the APMS 2019, Austin, TX, USA, 1–5 September 2019*; Ameri, F., Stecke, K., von Cieminski, G., Kiritsis, D., Eds.; IFIP Advances in Information and Communication Technology; Springer: Cham, Switzerland, 2019; Volume 566, pp. 615–623. [CrossRef]
53. Niermann, D.; Doernbach, T.; Petzoldt, C.; Isken, M.; Freitag, M. Software framework concept with visual programming and digital twin for intuitive process creation with multiple robotic systems. *Robot. Comput. Integr. Manuf.* **2023**, *82*, 102536. [CrossRef]
54. Clement, S.J.; McKee, D.W.; Romano, R.; Xu, J.; Lopez, J.M.; Battersby, D. The Internet of Simulation: Enabling agile model based systems engineering for cyber-physical systems. In Proceedings of the 2017 12th System of Systems Engineering Conference (SoSE), Waikoloa, HI, USA, 18–21 June 2017; pp. 1–6. [CrossRef]
55. Vereycken, Y.; Ramioul, M.; Hermans, M. Old wine in new bottles? Revisiting employee participation in Industry 4.0. *New Technol. Work. Employ.* **2021**, *36*, 44–73. [CrossRef]
56. Frysak, J.; Krenn, F.; Kaar, C.; Stary, C. Decision-making support for view-oriented I4.0 system architecture design. In Proceedings of the 2018 IEEE 20th Conference on Business Informatics (CBI), Vienna, Austria, 11–14 July 2018; pp. 186–195. [CrossRef]
57. Salehi, V. Development of an agile concept for MBSE for future digital products through the entire life cycle management called Munich Agile MBSE Concept (MAGIC). *Comput. Aided Des. Appl.* **2020**, *17*, 147–166. [CrossRef]
58. Salehi, V.; Wang, S. Munich agile MBSE concept (MAGIC). In Proceedings of the 22nd International Conference on Engineering Design (ICED19), Delft, The Netherlands, 5–8 August 2019; pp. 3701–3710. [CrossRef]
59. Mrugalska, B.; Ahmed, J. Organizational Agility in Industry 4.0: A Systematic Literature Review. *Sustainability* **2021**, *13*, 8272. [CrossRef]
60. Kayabay, K.; Gökalp, M.O.; Eren, P.E.; Koçyiğit, A. A workflow and cloud based service-oriented architecture for distributed manufacturing in Industry 4.0 context. In Proceedings of the 2018 IEEE 11th International Conference on Service-Oriented Computing and Applications, Paris, France, 20–22 November 2018; pp. 88–92. [CrossRef]
61. Santos, N.; Ferreira, N.; Machado, R.J. Towards agile architecting: Proposing an architectural pathway within an Industry 4.0 project. In *Information Systems: Research, Development, Applications, Education, Proceedings of the SIGSAND/PLAIS 2019, Gdansk, Poland, 19 September 2019*; Wrycza, S., Maślankowski, J., Eds.; Lecture Notes in Business Information Processing; Springer: Cham, Switzerland, 2019; Volume 359, p. 359. [CrossRef]
62. Stachowiak, H. *Allgemeine Modelltheorie*; Springer: Wien, Austria; New York, NY, USA, 1973.
63. Kannengiesser, U.; Gero, J.S. What distinguishes a model of systems engineering from other models of designing? An ontological, data-driven analysis. *Res. Eng. Des.* **2022**, *33*, 129–159. [CrossRef]
64. Haunschmied, D.; Kannengiesser, U. How are Industry 4.0 reference architectures used in CPPS development? In Proceedings of the 2022 25th Euromicro Conference on Digital System Design (DSD), Maspalomas, Spain, 31 August–2 September 2022; pp. 641–648. [CrossRef]
65. Loch, F.; Vogel-Heuser, B.; Reinhold, F.; Böck, S.; Hofer, S.; Reiss, K. Investigating mental models of mechanical engineering students. In Proceedings of the 2019 18th International Conference on Information Technology Based Higher Education and Training (ITHET), Magdeburg, Germany, 26–27 September 2019; pp. 1–6. [CrossRef]
66. Whitcomb, C.A.; Auguston, M.; Giammarco, K. Composition of behavior models for systems architecture. In *Modeling and Simulation Support for System of Systems Engineering Applications*; Rainey, L.B., Tolk, A., Eds.; John Wiley & Sons Ltd.: Chichester, UK, 2015; pp. 361–391. [CrossRef]
67. Keil, F.C.; Lockhart, K.L. Beyond cause: The development of clockwork cognition. *Curr. Dir. Psychol. Sci.* **2021**, *30*, 167–173. [CrossRef]
68. McCarthy, A.M.; Keil, F.C. A right way to explain? Function, mechanism, and the order of explanations. *Cognition* **2023**, *238*, 105494. [CrossRef]
69. Moody, D.L. The "physics" of notations: Toward a scientific basis for constructing visual notations in software engineering. *IEEE Trans. Softw. Eng.* **2009**, *35*, 756–778. [CrossRef]

70. Gupta, R.; Jansen, N.; Regnat, N.; Rumpe, B. Design guidelines for improving user experience in industrial domain-specific modelling languages. In Proceedings of the 25th International Conference on Model Driven Engineering Languages and Systems: Companion Proceedings (MODELS '22), Association for Computing Machinery, New York, NY, USA, 23 October 2022; pp. 737–748. [CrossRef]
71. Gruber, T.R. Toward principles for the design of ontologies used for knowledge sharing? *Int. J. Hum. Comput. Stud.* **1995**, *43*, 907–928. [CrossRef]
72. International Electrotechnical Commission. *IEC 61131-3*; Programmable Controllers—Part 3: Programming Languages. International Electrotechnical Commission: Geneva, Switzerland, 2013.
73. International Electrotechnical Commission. *IEC 61499-1*; Function Blocks—Part 1: Architecture. International Electrotechnical Commission: Geneva, Switzerland, 2012.
74. Wohlrab, R.; Pelliccione, P.; Knauss, E.; Larsson, M. Boundary objects and their use in agile systems engineering. *J. Softw. Evol. Process* **2018**, *31*, e2166. [CrossRef]
75. Fehrenbach, D.; Razek, A.R.A.; van Husen, C. Developing a rapid service prototyping framework. In Proceedings of the 2022 IEEE 28th International Conference on Engineering, Technology and Innovation (ICE/ITMC) & 31st International Association for Management of Technology (IAMOT) Joint Conference, Nancy, France, 19–23 June 2022; pp. 1–6. [CrossRef]
76. Kannengiesser, U.; Krenn, F.; Stary, C. A Situated Cognition Model for CPPS Testing. In Proceedings of the 2021 4th IEEE International Conference on Industrial Cyber-Physical Systems (ICPS), Victoria, BC, Canada, 10–12 May 2021; pp. 207–212. [CrossRef]
77. Baldwin, C.Y.; Clark, K.B. *Design Rules: Volume 1. The Power of Modularity*; MIT Press: Cambridge, MA, USA, 2000.
78. Lukyanenko, R.; Parsons, J.; Storey, V.C.; Samuel, B.M.; Pastor, O. Principles of Universal Conceptual Modeling. In *Enterprise, Business-Process and Information Systems Modeling, Proceedings of the BPMDS EMMSAD 2023, Zaragoza, Spain, 12–13 June 2023*; Van der Aa, H., Bork, D., Proper, H.A., Schmidt, R., Eds.; Lecture Notes in Business Information Processing; Springer: Cham, Switzerland, 2023; Volume 479, pp. 169–183. [CrossRef]
79. Thalheim, B. Towards a theory of conceptual modelling. *J. Univers. Comput. Sci.* **2010**, *16*, 3102–3137. [CrossRef]
80. Jackson, A.; Palmer, B.; Cobb, D.; McCreless, J. Model of models methodology: Reuse your architectural data. *INCOSE Int. Symp.* **2021**, *31*, 1303–1318. [CrossRef]
81. Zuefle, M.; Krause, D. Multi-disciplinary product design and modularization—Concept introduction of the Module Harmonization Chart (MHC). *Procedia CIRP* **2023**, *119*, 938–943. [CrossRef]
82. Hasselbring, W.; Wojcieszak, M.; Dustdar, S. Control flow versus data flow in distributed systems integration: Revival of flow-based programming for the industrial internet of things. *IEEE Internet Comput.* **2021**, *25*, 5–12. [CrossRef]
83. Kvan, T. Collaborative design: What is it? *Autom. Constr.* **2000**, *9*, 409–415. [CrossRef]
84. Kannengiesser, U.; Neubauer, M.; Heininger, R. Subject-oriented BPM as the glue for integrating enterprise processes in smart factories. In *On the Move to Meaningful Internet Systems: OTM 2015 Workshops, Proceedings of the Confederated International Workshops: OTM Academy, OTM Industry Case Studies Program, EI2N, FBM, INBAST, ISDE, META4eS, and MSC 2015, Rhodes, Greece, 26–30 October 2015*; Ciuciu, I., Panetto, H., Debruyne, C., Aubry, A., Bollen, P., Valencia-García, R., Mishra, A., Fensel, A., Ferri, F., Eds.; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2015; Volume 9416, pp. 77–86. [CrossRef]
85. Neubauer, M.; Stary, C. (Eds.) *S-BPM in the Production Industry: A Stakeholder Approach*; Springer Nature: Cham, Switzerland, 2017. [CrossRef]
86. Stary, C.; Elstermann, M.; Fleischmann, A.; Schmidt, W. Behavior-centered digital-twin design for dynamic cyber-physical system development. *Complex Syst. Inform. Model. Q. (CSIMQ)* **2022**, *30*, 31–52. [CrossRef]
87. Heininger, R.; Jost, T.E.; Stary, C. Enriching socio-technical sustainability intelligence through sharing autonomy. *Sustainability* **2023**, *15*, 2590. [CrossRef]
88. INCOSE. *Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities*, 4th ed.; International Council on Systems Engineering: San Diego, CA, USA, 2015.
89. Pahl, G.; Beitz, W. *Engineering Design: A Systematic Approach*; Springer: Berlin/Heidelberg, Germany, 2007.
90. Kannengiesser, U.; Oppl, S. Business processes to touch: Engaging domain experts in process modelling. In *BPM 2015*; CEUR-WS: Innsbruck, Austria, 2015.
91. Fleischmann, C. A tangible modeling interface for subject-oriented business process management. In *S-BPM in the Wild: Practical Value Creation*; Fleischmann, A., Schmidt, W., Stary, C., Eds.; Springer: Cham, Switzerland, 2015; pp. 135–151. [CrossRef]
92. Moattar, H.; Bandara, W.; Kannengiesser, U.; Rosemann, M. Control flow versus communication: Comparing two approaches to process modelling. *BPMJ* **2022**, *28*, 372–397. [CrossRef]
93. Moser, C.; Kannengiesser, U.; Elstermann, M. Examining the PASS approach to process modelling for digitalised manufacturing: Results from three industry case studies. *EMISAJ* **2022**, *17*, 1–24. [CrossRef]
94. Börger, E.; Stärk, R. *Abstract State Machines: A Method for High-Level System Design and Analysis*; Springer: Berlin/Heidelberg, Germany, 2003.
95. Müller, H. Using S-BPM for PLC code generation and extension of subject-oriented methodology to all layers of modern control systems. In *S-BPM ONE—Scientific Research*; Stary, C., Ed.; LNBIP 104; Springer: Berlin/Heidelberg, Germany, 2012; pp. 182–204. [CrossRef]

96. Zoitl, A.; Lewis, R. *Modelling Control Systems Using IEC 61499*, 2nd ed.; The Institution of Engineering and Technology: London, UK, 2014.
97. Kannengiesser, U.; Krenn, F.; Stary, C. A behaviour-driven development approach for cyber-physical production systems. In Proceedings of the 2020 IEEE Conference on Industrial Cyberphysical Systems (ICPS), Tampere, Finland, 10–12 June 2020; pp. 179–184. [CrossRef]
98. Oppl, S. Articulation of work process models for organizational alignment and informed information system design. *Inf. Manag.* **2016**, *53*, 591–608. [CrossRef]
99. Kannengiesser, U.; Frysak, J.; Stary, C.; Krenn, F.; Müller, H. Developing an engineering tool for Cyber-Physical Production Systems. *Elektrotechnik Informationstechnik* **2021**, *138*, 330–340. [CrossRef]
100. Mandel, C.; Kaspar, J.; Heitmann, R.; Horstmeyer, S.; Martin, A.; Albers, A. Implementation and assessment of a comprehensive model-based systems engineering methodology with regard to user acceptance in practice. *Procedia CIRP* **2023**, *119*, 897–902. [CrossRef]
101. Mileva-Boshkoska, B.; Rončević, B.; Uršič, E.D. Modeling and Evaluation of the Possibilities of Forming a Regional Industrial Symbiosis Networks. *Soc. Sci.* **2018**, *7*, 13. [CrossRef]
102. Polančič, G.; Orban, B. An experimental investigation of BPMN-based corporate communications modeling. *Bus. Process Manag. J.* **2023**, *29*, 1–24. [CrossRef]
103. Fink, A.; Schlake, O. Scenario management—An approach for strategic foresight. *Compet. Intell. Rev.* **2000**, *11*, 37–45. [CrossRef]
104. Lehner, D.; Sint, S.; Eisenberg, M.; Wimmer, M. A pattern catalog for augmenting Digital Twin models with behavior. *At-Automatisierungstechnik* **2023**, *71*, 423–443. [CrossRef]
105. Graessler, I.; Hentze, J. The new V-Model of VDI 2206 and its validation. *At-Automatisierungstechnik* **2020**, *68*, 312–324. [CrossRef]