

Article

# Geo-Spatial Context Provision for Digital Twin Generation

Thomas Ernst Jost , Christian Stary  and Richard Heininger 

Institute of Business Informatics—Communications Engineering, Johannes Kepler University Linz,  
Altenberger Straße 69, 4040 Linz, Austria

\* Correspondence: thomas.jost@jku.at

**Abstract:** Light detection and ranging technology allows for the creation of detailed 3D point clouds of physical objects and environments. Therefore, it has the potential to provide valuable information for the operation of various kinds of cyber-physical systems that need to be aware of, and interact with, their surroundings, such as autonomous vehicles and robots. Point clouds can also become the basis for the creation of digital representations of different assets and a system's operational environment. This article outlines a system architecture that integrates the geo-spatial context information provided by LiDAR scans with behavioral models of the components of a cyber-physical system to create a digital twin. The clear distinction between behavior and data sets the proposed digital twin architecture apart from existing approaches (that primarily focus on the data aspect), and promotes contextual digital twin generation through executable process models. A vaccine logistics automation use case is detailed to illustrate how information regarding the environment can be used for the operation of an autonomous robot carrying out transport preparation tasks. Besides supporting operation, we propose to combine context data retrieved from the system at different points in the logistics process with information regarding instances of executable behavior models as part of the digital twin architecture. The twin can subsequently be used to facilitate system and process monitoring through relevant stakeholders and structure context data in a user-centric fashion.

**Keywords:** digital twin architecture; logistics; LiDAR technology; behavior specification; process modeling & execution



**Citation:** Jost, T.E.; Stary, C.; Heininger, R. Geo-Spatial Context Provision for Digital Twin Generation. *Appl. Sci.* **2022**, *12*, 10988. <https://doi.org/10.3390/app122110988>

Academic Editor: Dimitris Mourtzis

Received: 21 September 2022

Accepted: 26 October 2022

Published: 30 October 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

A core component of Cyber-Physical Systems (CPS) are sensors to obtain information about the physical world [1]. They enable capturing highly relevant environment data, including geo-spatial context information, to successfully operate an embodied CPS. Light detection and ranging (LiDAR) technology has been used to create detailed scans of physical objects, which can become the basis for digital representations of these objects. LiDAR technology is, e.g., used to capture real-time data concerning the surroundings of autonomous vehicles to detect and recognize objects [2]. It is also applied in the context of indoor navigation of autonomous systems for the complex task of simultaneous localization and mapping (SLAM) [3]. Scan data has also become a crucial building block for the creation of digital twins [4].

Digital twins (DTs) are traditionally seen as simulated representations of a physical object or a set of objects (cf. one of the earliest definitions of the term provided by the NASA in [5]). Currently, digital twins are often used in the context of industry to represent objects that are part of production (production systems, material, products, ...). Especially in the context of Industry 4.0 and smart manufacturing, digital twins are seen as a promising technology—cf. [6] with application areas pertaining to product design, production, and prognostics and health management. Digital twins can serve as a design and engineering prototype for physical objects and allow for testing of different functionality [7]. In the context of CPS, executable behavior specifications in the form of subject-oriented process

models were proposed for exploring and validating CPS behavior in the form of a digital twin thread [8,9].

Digital twins aim to serve as a single hub of information about the real-world object(s) for a user, linking systems and realizing the structuring, monitoring, and exploitation of data [10]. Therefore, structuring the data produced by a CPS during its operation is a crucial aspect of digital twins to enable the monitoring of the system and its environment while it executes its behavior in the respective CPS context, e.g., business processes. To this end, the information needs to be structured in a form that makes it accessible to the relevant stakeholders, leading to usage-driven deployment of digital twins and thus user acceptance [11].

Although geo-spatial information is considered to provide useful context data, one challenge concerns deriving relevant information from the created raw scan data (i.e., 3D point clouds). In this context, semantic segmentation, i.e., classifying every point in a scene and assigning it a label to give the data semantic meaning, has been tackled extensively in the literature through a variety of approaches, recently propelled forward by deep learning (see [12]). Performing such segmentation tasks is often a part of the early stages in the digital twinning process (see [13,14]). Different formats for 3D models have furthermore been proposed in a variety of different contexts to structure the relevant produced information (e.g., data formats for digital city models (see, e.g., the City Geography Markup Language (CityGML) at <https://www.ogc.org/standards/citygml>, retrieved 8 June 2022) that highlight buildings, vegetation, ...).

A further challenge concerns the (real-time) visualization of created digital representations (cf. [15]). Depending on the intended DT use, the produced geo-spatial context needs to be potentially integrated with other information to serve a specific purpose (see the DT created in [16], which incorporates data from different auxiliary sources). In the context of CPS, the produced geo-spatial data needs to be linked with other information, e.g., be embedded with other sensor data, since one of the core added value propositions of DTs lies in making use of linked information [10].

In this article, we explore the structuring of geo-spatial information, as derived from 3D point cloud data, in order to consolidate it with relevant information required for designing and running digital twins as digital threads (cf. [17]). Hence, we presume that the raw 3D scan data has already been processed to be further utilized as geo-spatial context information. Focusing on geo-spatial context integration requires an appropriate representation and visualization of this information [18–20].

In cyber-physical settings, scans conducted with LiDAR technology are used to capture the operational environment of CPS, and to enable a CPS to accomplish its tasks as part of a business case or process [21]. We showcase how a wider view of the concept of digital twins, where not only data about the physical object, but also its behavior is captured, can facilitate the structuring of scan data produced alongside the operation of the system in a way that it becomes better accessible to relevant stakeholders. Such a conception of digital twins has recently emerged in CPS research (see [8]) and has potential for structuring a variety of data produced during operation. Process modeling (which is the basis for such a kind of behavioral twins) was also identified as a currently emerging research direction in a recent bibliometric analysis of the field of digital twins [22]. We contribute to this new research trend by providing a digital twin architecture based on subject-oriented process models [23].

We exemplify a logistics showcase of a digital process twin architecture and demonstrate its models' potential for context-sensitive geo-spatial applications. The behavioral specification of the CPS is created utilizing subject-oriented process models. These (executable) models are used as a basis to structure scan data alongside the conducted behavior of the CPS (components). Compared to simply structuring scan data in the form of a time series, using process models has the advantage of associating produced information directly with the behavior of a system at a certain point in time. This additional context enriches the digital twin with a behavior perspective for each system component and the

components' interplay representing the overall system behavior. Once this information can be conveyed to relevant stakeholders, the environmental context at different points in time can facilitate the integration of cyber-physical systems into business operation. This context can be particularly considered for supporting decision makers and process monitors.

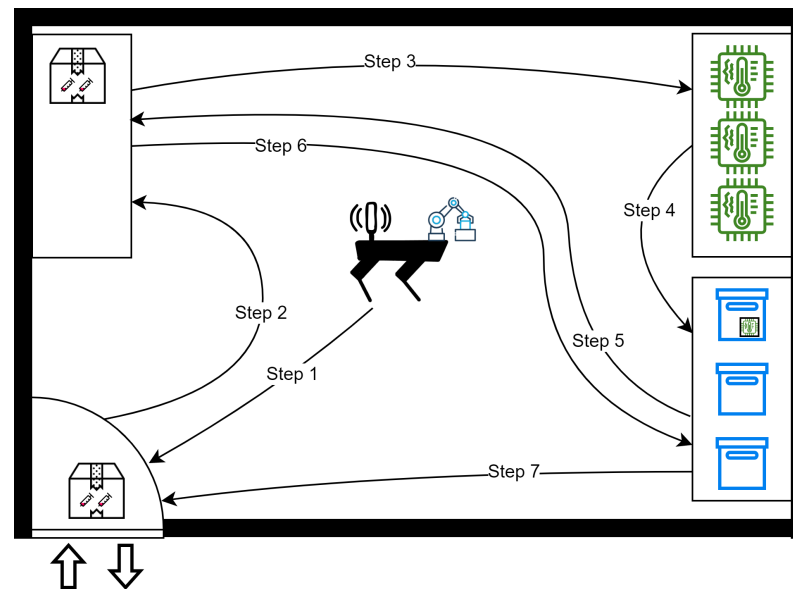
The article is structured as follows: Section 2 introduces the vaccine logistics use case, giving insight into the general workflow, and how and when LiDAR technology can be applied to obtain geo-spatial context information. Based on this scenario, Section 3 provides the proposed digital twin architecture and exemplifies models of LiDAR-relevant parts of the logistics process. In particular, the integration of behavior models with scan data is explored. Along with the architecture, a methodology is introduced to guide implementing the proposed architecture. An application for behavior monitoring is also outlined. Section 4 highlights related work. First, the use of LiDAR technology and the produced scan data in digital twin research is examined. The proposed architecture is positioned in relation to existing approaches and its difference and potential are detailed. Secondly, methodologies for the creation of digital twins are explored to reflect on the presented methodological guidance and to identify topics of future research with respect to development procedures. Section 5 discusses in how far the knowledge base on subject-oriented design and engineering could be enriched. Finally, Section 6 briefly summarizes the presented work and discusses the results. Limitations and possibilities for future investigations are outlined.

## 2. CPS-Based Vaccine Logistics

The use case that we utilize throughout this article was inspired by issues concerning the distribution of vaccines in the context of COVID-19, with different types of vaccine requiring different transport and storage conditions. It pertains to supporting the handling and packaging of vaccines as part of the distribution process (during the “last mile”, i.e., local redistribution) through automation and IoT technologies. The result is a CPS encompassing different components. To support the monitoring of transport conditions, transport boxes are dynamically equipped with sensors, depending on the specific requirements of the payload (i.e., the vaccine). Examples of potential sensors include temperature sensors for vaccines that require constant cooling to a certain temperature (as is the case with the COVID-19 vaccine of Pfizer-BioNTech), humidity sensors to detect potential spills, and shock sensors to indicate potential damage. Transport boxes equipped with such sensors can be used for real-time monitoring of conditions, with abnormalities triggering immediate interventions, such as notifications of the relevant stakeholders. They can similarly be used for auditing purposes, to showcase the integrity of transported payloads. The tasks necessary for preparing the boxes are automated and carried out by a robot. This robot is equipped with an arm unit for interaction with its environment and a scanner to sense and assess its surroundings. For showcasing the proposed digital twin architecture and the related concepts, the specific type of robot to be used is not important (due to the used abstraction mechanisms that will be detailed further below), as long as it can execute the required behavior and provide the required data. For a prototype set-up that is in the works at the time of writing this article, Boston Dynamic's robot dog Spot (see <https://www.bostondynamics.com/products/spot>, retrieved 8 June 2022) is used, due to availability and since there exist options for the required periphery (see <https://www.bostondynamics.com/sites/default/files/inline-files/spot-arm.pdf> and <https://www.bostondynamics.com/sites/default/files/inline-files/spot-eap.pdf>, retrieved 8 June 2022). It should be noted that in an industrial warehouse setting, there exist potentially cheaper options that can provide the necessary features for implementing such a set-up, such as various Automated Guided Vehicles (AGVs). The environment the robot operates in includes three types of shelves that, respectively, house vaccine packages, the transport boxes, and available sensors.

### 2.1. General Workflow

The set-up of the scenario and the minimal number of involved steps for transport preparation can be seen in Figure 1. The process starts with a package containing vaccines arriving at the intended input location (in our simplified set-up this would be the door of the room). A photoelectric barrier sensor that is part of the room registers the arrival. This triggers the further handling of the package through the robot.



**Figure 1.** Vaccine logistics illustration case - Schematic of the system set-up and the minimal number of involved steps.

**(Step 1)** The robot moves to the location of the package and reads a label attached to it, which provides the necessary information for the following steps (i.e., the type of vaccine and the specific transport assignment, and following from that the sensors that are required). The robot picks up the package.

**(Step 2)** The robot moves the package to the shelf on the left side of the room for intermediate storage.

**(Step 3)** The robot moves to the shelf on the upper right side of the room, which contains a selection of sensors to equip a transport box with. Based on the vaccine-specific information read from the label previously, the robot picks up a sensor (due to the capabilities of the arm only one sensor can be carried at a time).

**(Step 4)** The robot moves over to the shelf housing the transport boxes on the lower left of the room. The robot uses the arm unit to insert the sensor into the intended cavity on the side of the transport box. Upon successful insertion, the sensor is activated and begins actively sensing its environment. Sensor information can be accessed by the transport box, which can execute different actions based on the read values and additional information concerning the payload and transport assignment. In case further sensors are required, the robot returns to the shelf containing the sensors and repeats the preceding step, until the box is equipped with all the necessary sensors.

**(Step 5)** Once a box has been successfully prepared, the robot moves over to the shelf on the left side of the room and picks up the vaccine package intended for the transport container.

**(Step 6)** The robot transports the package to the shelf containing the transport boxes and puts it into the box it prepared in the previous steps. The robot connects to the box (which has network capabilities) to share information. This includes information about the payload and the transport assignment (influencing which actions the box takes in response to certain sensor readings) and the sensors that were equipped. When the box is notified that the preparation steps have been completed successfully and receives the corresponding

information, it closes itself automatically and informs the robot with some confirmation to be ready for transport.

**(Step 7)** Finally, the robot picks up the box, moves it to the output destination, and places it there.

Abstracting the behavior of the robot, it can be seen as a loop repeating the following two actions for each step: move to way-point and interact with environment.

## 2.2. Using LiDAR Technology for Obtaining Geo-Spatial Context

Assuming the intended use case, the layout of the room is known in advance. Thus, the positions of the different navigation way-points in the room and the locations that objects are picked up from or placed down at are known (with the robot keeping track of the status of each shelf position). The movement of the robot between the different way-points can thus be pre-planned, allowing for the avoidance of complex SLAM tasks.

For situations requiring the robot to actively be aware of contextual information pertaining to its surroundings, the scanner equipped on the robot is used to obtain the current geo-spatial status of its operational environment. This allows to adapt the robot's behavior if objects are not positioned as they should be (or missing altogether) or if obstacles appeared in its path.

During the movement between the different way-points an unexpected obstacle could block the robot. In this case, the robot is required to navigate autonomously around it, or to notify the person in charge of overseeing the process so that they may intervene. Similarly, when reaching a way-point, such as standing before a shelf, adjustments in the robots positioning could be needed to ensure that it is in the right position to interact with an object that is not placed properly. These adjustments can then be computed based on scan data. Different object detection approaches for point clouds have been proposed in the literature and can be adapted for such a task (see, e.g., approaches such as PointNet [24] and its successor PointNet++ [25], VoxelNet [26], or the approach described in [27]). Once objects in the robot's environment have been detected and classified, their distance and position in relation to the robot can be computed to determine its actions for re-positioning itself to be in the right place to interact with the object. In case re-positioning proves impossible (due to an object being placed in such a way that the robot cannot align itself to execute the intended interactions), a human needs to be called for assistance and restoring the operational environment.

## 3. A Digital Twin Architecture Linking CPS Behavior and Produced Operational Data

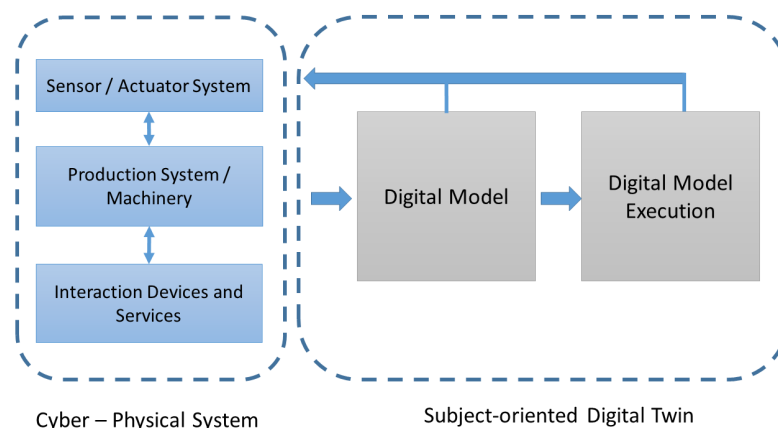
We now use the case described in the previous section as the basis for specifying the proposed digital twin architecture. The CPS of the illustration scenario encompasses a variety of different components. These components showcase different behavior and interact with each other as part of the transport preparation process. In this section, we first introduce the used conception of behavior-centered digital twins that can capture these aspects of a CPS. We briefly outline subject-oriented modeling as an approach for the creation of digital behavior models. Following that, we showcase models for a part of the process of the illustration case. Then, we detail the stakeholder-centered structuring of environmental context data through the process models. For the sake of run-time engine independence, we provide an abstract description of the required run-time behavior to execute the digital behavior models. Subsequently, we integrate the introduced concepts into an architecture proposal and introduce a methodology as guidance for realization. We also discuss some of the associated challenges and considerations. Finally, we exemplify an application and reveal benefits when using the architecture in practice for the case.

### 3.1. Behavior-Centered Digital Twins

We build upon the behavior-centered conception of digital twins of CPS as it was outlined in [8], with subject-oriented process modeling [23] as an approach for creating behavior specifications. This particular understanding of digital twins is depicted in



Figure 2. As outlined in [8], besides the digital model of the CPS, run time support connected to the cyber-physical components is available for model execution. This allows for changes to the digital model during operation that propagate to the components executing the behavior (and thus the model). The authors refer to this aspect as “design-integrated engineering”, entailing that the models of the CPS are subsequently refined to the point that they can be executed. As can be seen in the figure, data (such as collected sensor data) also flows back from the CPS to its digital representation. Therefore, this digital twin conception also provides an infrastructure for the monitoring of CPS component behavior and is used as the basis for integrating geo-spatial context.



**Figure 2.** The digital twin thread concept for a CPS, based on modeling and execution, reprinted from (Ref. [8], p. 39).

Now we specify the central elements of subject-oriented process models, as they are relevant for the digital twin architecture that utilize them. Subject-oriented modeling accounts for some of the unique characteristics of CPS, such as allowing to depict heterogeneous components in a unifying way by focusing on their behavior [8,28]. Subject-oriented modeling originated from the structure of natural language, specifically the components of sentences, i.e., subject, predicate, and object [23]. This allows for modeling inspired by natural language sentence formulation. Furthermore, the minimal number of symbols used by the associated standard modeling notation makes it more easily accessible. This is indicated by a recent research effort [29] to empirically examine the properties of both a control flow modeling paradigm (e.g., BPMN) and a communication modeling paradigm (e.g., subject-oriented process models).

The subject-oriented approach uses two types of models to specify processes [23]: Subject Interaction Diagrams (SIDs) and Subject Behavior Diagrams (SBDs). SIDs depict different subjects and their interactions with each other as part of a process. These interactions are modeled as messages that are exchanged between the different subjects. In this kind of depiction, a subject is understood as an encapsulation of a certain behavior that it executes as part of the process, so each subject in the SID has an associated SBD. Different subject behavior is executed in parallel, with messages serving to synchronize behavior. The instantiation of the subject is furthermore left open during the modeling process. Whether a subject behavior is executed by a human, an organization, a software component, a cyber-physical component, etc. is decided at a later point in time. SBDs now depict behavior as a sequence of states, including function states (performing some local action), send states (sending a message to another subject), and receive states (receiving a message from another subject). Furthermore, one state needs to be marked as the start state of the sequence and there needs to be at least one end state (multiple end states are possible). States are connected through transitions (usually notated as arrows labeled with the result of the previous state). Different tools have been proposed that support the creation, validation, and execution of subject-oriented models. Such run-time engines originate from

both the commercial (see the suites Metasonic (<https://www.metasonic.de/en/>, retrieved 9 June 2022) and Compunity (<https://compunity.eu/>, retrieved 9 June 2022)) and academic sector (e.g.,: [30,31]).

Subject-oriented business processes are embedded in organizations within the framework of Subject-oriented Business Process Management (S-BPM) [23]. Thereby, the processes are considered in different phases, which are called activity bundles. These activity bundles can be run sequentially in an iterative way, as shown in Figure 3. However, since some activity bundles can also be skipped or used within an arbitrary sequence due to the self-contained nature of each step, the methodological frame has also been termed “open control cycle” [23]. The activity bundles address the analysis, modeling, validation, optimization, organization-specific implementation, IT-implementation, and monitoring of processes. In the course of analysis, all process-relevant information is captured, while in modeling this information is brought into a subject-oriented process model. Consequently, the activity bundles of analysis and modeling are essentially about which subjects perform which activities on which objects utilizing which tools, and in which way the subjects interact to achieve the desired process goals and results. Validation verifies that the specified process is effective, i.e., that it produces the expected output in the form of a product or service. Optimization means finding an optimal design of a process with regard to process parameters such as duration, costs and frequency. Validated and optimized processes are embedded in the organization during organization-specific implementation, and this includes any adaptation of the process and organizational structure that represent the social environment. The IT implementation of a process means mapping it as an IT-supported workflow by integrating a suitable user interface, the flow logic and the IT systems involved. Ongoing monitoring gathers measurement data during process execution and can be used, for example, to calculate the actual values for the key performance indicators defined during analysis and modeling [23].

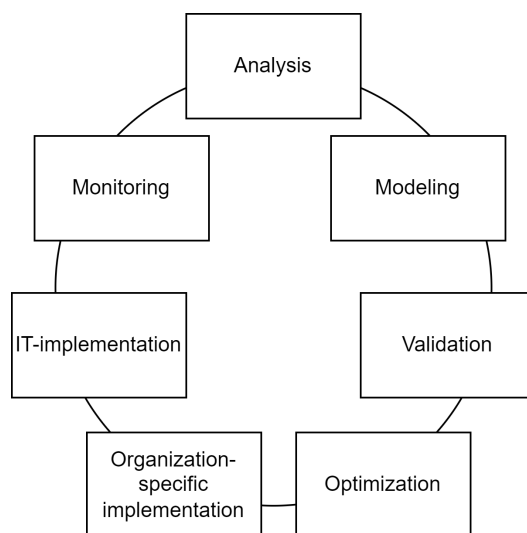
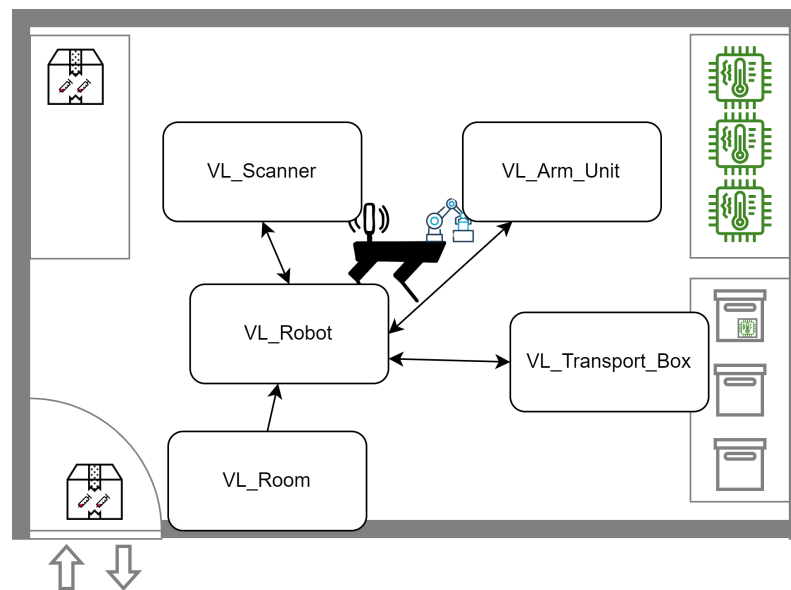


Figure 3. Open Control Cycle of S-BPM Activity Bundles, reprinted from (Ref. [23], p. 31) .

### 3.2. Sample Model Variants for the Logistic Use Case

The general modeling approach is depicted in Figure 4 based on the use case scenario described in the previous section. Each active component is modeled as its own subject. The connections between the subjects contain messages to implement the entire CPS behavior.



**Figure 4.** General modeling approach.

As part of the proposed digital twin architecture, subject-oriented models are used to depict the individual behavior of CPS components and their interactions with each other as part of different processes. Figure 5 shows the subject interaction diagram of the components involved in the transport preparation process on a high level of abstraction. All models were created using the Compunity suite, which uses slightly different terminology, e.g., component interaction diagram instead of subject interaction diagram and step instead of state. For the sake of understandability, we chose to utilize the more general terms common to subject-orientation consistently throughout this article. Figure 5 shows which messages flow between the different subjects that are part of the process, with the notification of the room triggering the robot's behavior, and the robot issuing commands to its attached add-ons (here modeled as separate subjects). The content of the messages is detailed in terms of an abstract data structure that needs to be derived from the respective use case. It can also be aligned with existing data models. Depending on the chosen level of abstraction, the behavior of the robot and its add-ons could be unified into a singular subject, or separated even further. As was mentioned in the previous sub-section, if the digital behavior models are used in the sense of design-integrated engineering, a refinement to a fine-grained level will be necessary to put them into operation. Considering that the digital twin should meet the requirements of its potential users, a high level of abstraction will still be useful in cases where detailed information on behavior is not needed and might overwhelm a CPS stakeholder.

To demonstrate the modeling of component behavior we detail the part of the transport preparation process concerning the selection of a sensor through the robot (step 4). The behavior of the robot, as described above, was now modeled in a SBD, as seen in Figure 6. It follows the general activities of "move to way-point" and "interact with environment" as outlined above. According to the SID in Figure 5, the arm unit and scanner were modeled as separate subjects with their encapsulated behavior. The behavior is again depicted on a high level of abstraction.



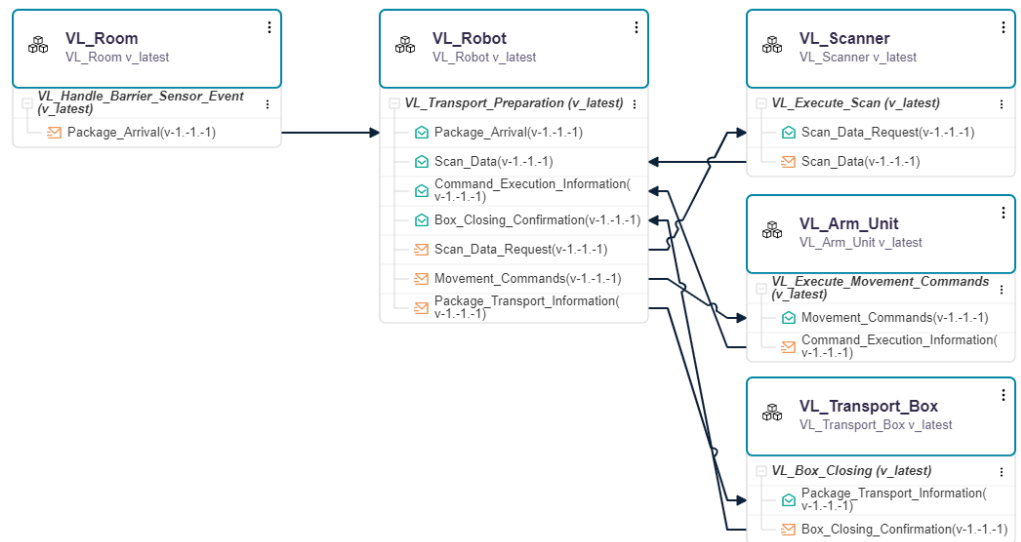


Figure 5. Vaccine logistics illustration case - Interaction between different components during transport preparation.

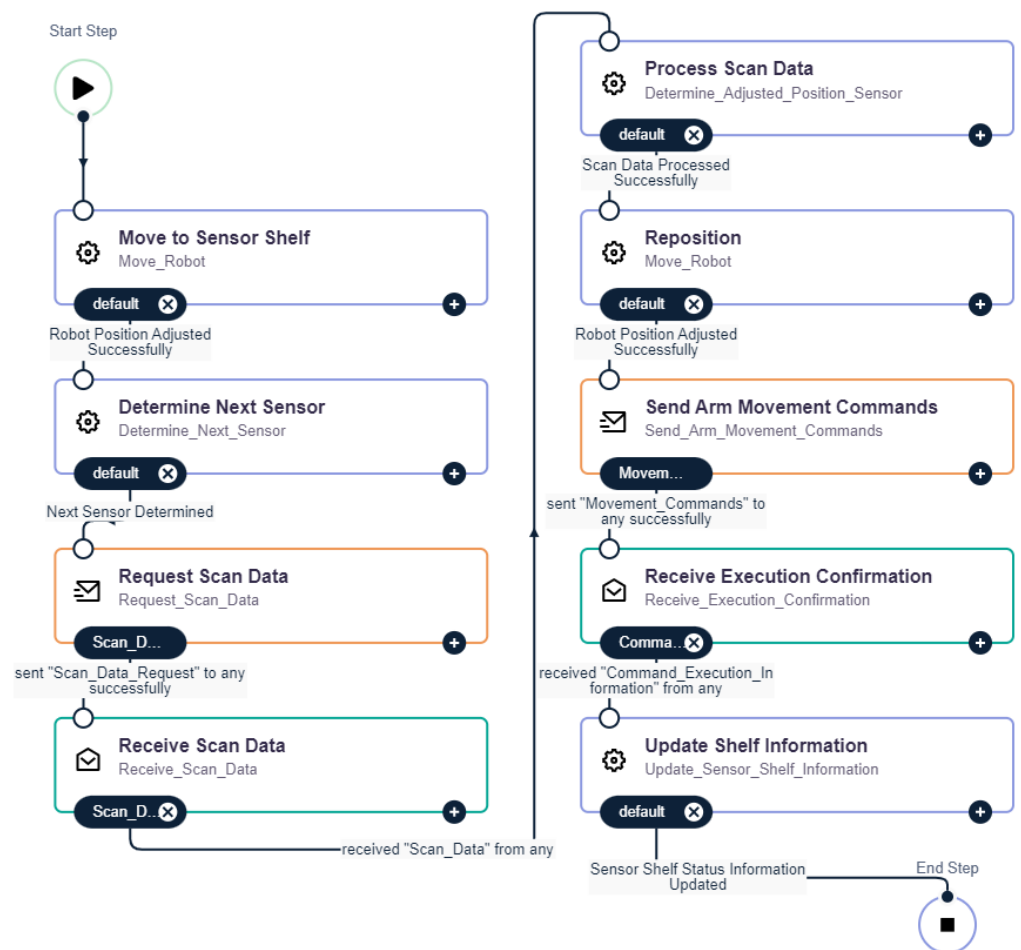


Figure 6. Vaccine logistics illustration case - Robot behavior during step 4 of transport preparation process (selection of a sensor).

### 3.3. Stakeholder-Centered Structuring of Environmental Context Data

In the described use case, the robot uses the scanner periodically as part of its tasks to assess its surroundings. The data provided by the scanner is used as a source for deriving

contextual information to dynamically adjust the robot's behavior to its environment. In the proposed digital twin architecture, another core purpose of these data are to save and integrate it with other information to create a representation of the CPS in a way that it supports process stakeholders. The results of scans can be periodically saved to create a timeline of how the operational environment of the CPS has changed, with the different objects (the robot, vaccine packages, sensors, transport boxes) moving/being moved and leaving/entering the area. From this information, a digital twin of the CPS environment is created that can be used to document and assess these environmental states and changes for different purposes (monitoring, auditing, ...). We have already described situations in Section 2.2 where process participants may need to access this information. There exist other information sources as well. The room outlined as part of our proposed set-up is equipped with different sensors, with the integration of the produced information promising to create an even more complete picture of the operational environment at certain points in time.

We propose the use of the behavioral twins of the CPS components outlined above (i.e., the executable, synchronized process models) to structure both the captured geo-spatial context and the information produced by other components. This integration is realized through associating the information provided by certain scans with the concrete steps in the process model during which these scans occur (or during which they are used) as an instance of the behavior model is executed synchronously with the system component showcasing the behavior.

We describe the envisioned structure of information for process model instances independently from details of concrete execution engines. The focus is on the basic elements of subject-oriented process models (specifically considering behavior diagrams) and using generic concepts for documenting the execution of processes. In the context of process monitoring (see [32] for the formal definitions of the terms used in this paragraph), various information regarding the execution of a process instance is generally recorded in the form of an event. An event was defined in [32] as a tuple  $(a, c, t, (d_1, v_1), \dots, (d_m, v_m))$ , with  $a$  being the activity name (in the case of subject-oriented models this would refer to a state of a SBD),  $c$  is the case identifier (i.e., referring to a process instance),  $t$  is the timestamp (in [33] both start and end timestamps are listed) and  $(d_1, v_1), \dots, (d_m, v_m)$  (where  $m \geq 0$ ) denote event attributes, with their names and values. The events that are generated by a process instance are referred to as a trace, with a so-called event log storing all completed traces pertaining to a process model.

These concepts already provide a framework for documenting relevant process data, such as timestamps for start and end times (cf. [33], where end times from event logs are used for remaining time prediction). The event attributes allow for the storage of various data produced during a process. Completed traces stored in the form of an event log document previous process instance executions. Furthermore, traces that are in the process of being recorded document an ongoing process instance execution.

By the activity name stored with an event, event data is set in relation to the model from which the process instance was created. It is assumed that states across the different SBDs of a process model have unique identifiers. Looking at the particularities of subject-oriented modeling, a few topics need to be considered for uniquely relating recorded events to states of SBDs. The same subject behavior can, e.g., be instantiated multiple times as part of process execution (Multi-Subjects, see <https://i2pm.net/wp-content/uploads/2020/04/20200223-Standardbuch-PASS.pdf#page=22>, retrieved 4 July 2022). In cases, where multiple instances of behavior modeled in a SBD are executed in parallel, this leads to potentially multiple event tuples with the same values for  $a$ ,  $c$ , and  $t$ . The addition of a SBD instance identifier allows to still uniquely relate events in such a case. It also supports the generation of traces not only on the level of whole process instances, but also on the level of instances of individual subject behavior part of the process instance. This way, the sequence of states that were performed (documented through events in a trace) can be showcased in relation to the options permitted in the models (one state in a SBD can occur multiple times as part of execution in case of a loop and certain modeled states might not occur

in case of exclusive paths). A graphical representation of current or completed process model instances with their performed states is still required for a stakeholder-centric use of event data.

Considering the digital twin thread concept outlined in Section 3.1, during each state instance (i.e., a state in the behavioral model that is currently being executed by a CPS component as part of a process instance), data can be provided by the system (component). So each state instance can potentially have associated data entries, as visualized in Figure 7 in the form of a simple UML class diagram. In this diagram, the data entry class is not further specified and left abstract, to reflect that different data can come from the system based on the scenario. For a CPS this can be sensor data (or other relevant data produced during the behavior of the system component), either captured or utilized during the respective state instance.

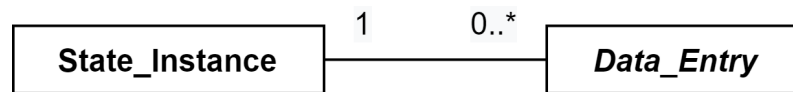


Figure 7. Each state instance can hold data entries provided by the system (component).

Utilizing the general concepts for documenting process executions outlined above, a state instance can be documented through an event, with the attributes  $(d_1, v_1), \dots, (d_m, v_m)$  of the event tuple holding the various data entries (with  $d$  denoting the data entry and  $v$  storing the data within the entry).

A very basic data entry containing a description of the data entry type, a timestamp, and the associated data needs to be put into relevant context, to depict data in a specific structure. This can be realized through inheriting and specifying the abstract class, with Figure 8 showcasing a few possible data entry types concerning sensor data from the illustration case (here depicted in UML through extending the abstract data entry class).

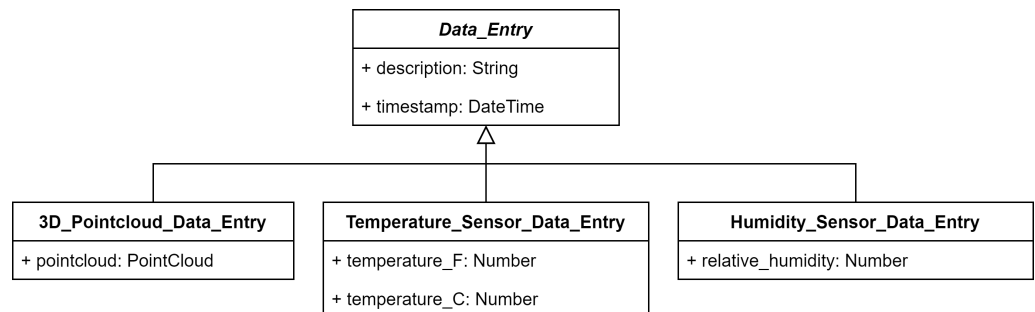


Figure 8. An outline of a minimal data structure for different possible types of data entries storing context information.

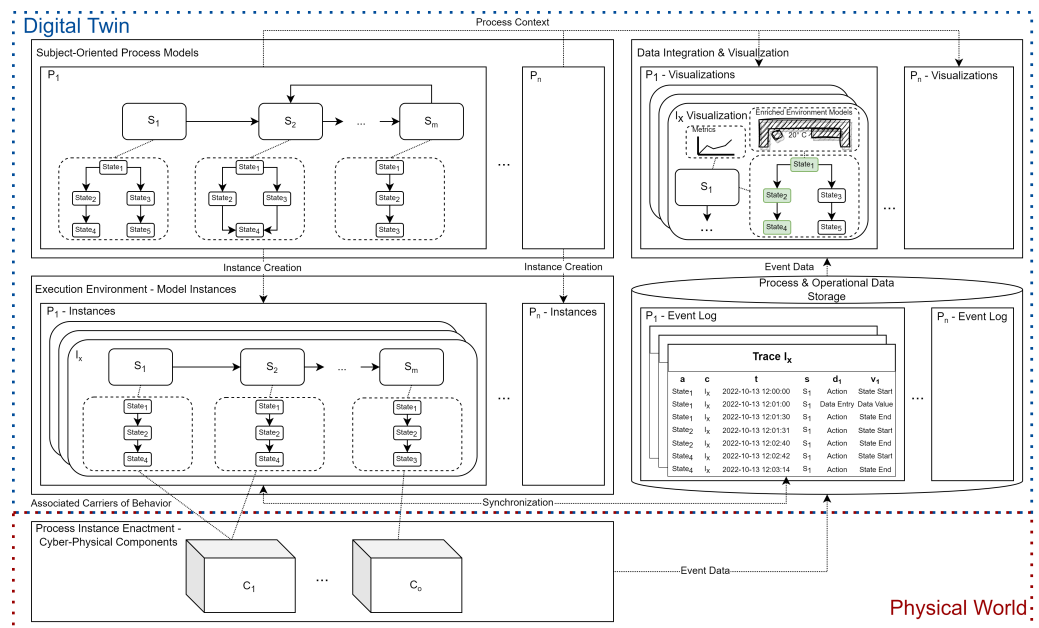
Concerning the enrichment of the behavioral model instances and their documentations with LiDAR-provided geo-spatial context information, we distinguish between different possibilities based on how much other context information concerning the CPS is directly integrated with the geo-spatial representation. The data structure highlighted in Figure 8 shows the lowest level of integration, where the raw 3D point cloud data is stored as part of one specific data entry type, and different types of context data have their own separate entry types. A higher level of semantic richness of geo-spatial context information can be achieved by storing a semantically segmented point cloud as part of a data entry. On higher levels of integration, data from previously separate data entry types is incorporated directly into a 3D representation of the operational environment, such as sensor readings being displayed next to the digital model of the physical sensor producing the data. Creating such a representation would require more computational effort. As part of the illustration case, processing of raw point cloud data is already required as part of CPS component behavior, since positional adaptations need to be realized based on detected object positions. The results can therefore be given to the digital twin and stored. In cases

where point cloud data is not already processed in some form as part of the behavior of the physical twin, the digital twin itself would have to provide the necessary functionality to generate representations meeting the requirements of DT stakeholders.

The organization of data outlined in this section allows a process stakeholder to view important information of the CPS associated with process model instances as they are currently being executed. A stakeholder can access data of the current state in the behavior model instance or of previous states that were already completed. In the case of geo-spatial context information, it also needs to be visualized in a way that facilitates the needs of process stakeholders. This aspect has been explored in the literature through the use of virtual reality technology and game engines (see, e.g., [34]). Information concerning the completed process model instance executions can furthermore be saved to document the system’s previous behavior and the environmental states and data associated with it.

### 3.4. Integrating the Introduced Concepts

The overall proposed approach to the creation of digital (process) twins, whose key concepts and elements were introduced in the previous sub-sections, is presented in Figure 9 in an integrated form. The graphic delineates the different involved elements, from the initial models depicting a number of business-relevant, technology-supported processes, to the created instances and their assigned cyber-physical behavior carriers producing event data. The event data is used to keep instances up-to-date with the state of the real-world processes. It is furthermore processed and integrated with the originally modeled processes to create different visualizations of the operational environment at different points in time, depending on stakeholder needs.



**Figure 9.** The proposed concept for the creation of digital (process) twins built on ideas from subject-oriented business process management to link behavior with geo-spatial context information.

For the realization of the introduced architecture, we furthermore propose a number of development steps, intended to serve as a general guideline. This methodology is based upon the general subject-oriented approach discussed in Section 3.1 and extends it towards incorporating the necessary steps to implement the architecture. It is also used as a framework to point out some of the challenges and considerations related to the required technology and design decisions. The basic assumption is that the system for which the architecture should be implemented is already known and specified to a certain degree.

- First, it is necessary to determine the set of organizational processes for which the twin could be implemented. Candidates are all processes that are, at least partially, enacted/supported by cyber-physical components.
- Next, it will be necessary to determine the overall goal and the envisioned use cases of the digital twin. This involves gathering stakeholder needs and requirements. These are used as input for the following steps and will determine various aspects of the final twin, such as behavior specification granularity and the data that needs to be gathered from the twinned system. Already, some attention should be given to the technical capabilities of components with regard to the provision of data (performance, network capabilities). Based on this information, processes are chosen from the candidate list.
- Subsequently, it is required to create the subject-oriented specification of the selected processes. The processes are already put in place and are executed as part of the organizational environment. Thus, if certain process documentations are already present (even if the used notation is not subject-oriented), they can be used as input. An example would be a natural language description, such as the illustration case outline that was used as input for creating the sample model variants in Section 3.2. Similar artefacts, such as documentations of control software and system models, can be used to gain knowledge about the behavior of cyber-physical components. There also exist techniques for the elicitation of stakeholder knowledge with regard to business processes. Due to the heterogeneous nature of CPS, this step will require inputs from many different domain experts to specify the process. This constitutes a major challenge that one needs to be aware of during architecture implementation. Once sufficient knowledge about the processes has been gathered (sufficiency will depend on many factors, such as the required level of process granularity), the modeling can start. Generally, one of the first steps of subject-oriented process modeling is to determine the subjects and the messages that they exchange. This is a question of decomposing a system and finding the appropriate level of abstraction. With regard to CPS, this entails that choices need to be made with regard to representing different components through their relevant behavior. Once subjects have been determined, their individual behavior is detailed in the form of an SBD. Finally, the models need to be validated through the stakeholders holding the relevant knowledge. Like the modeling itself, this step can be supported by tools that allow for static checking and interactive enactment of the models to ensure syntactic and semantic correctness. Any of the activities associated with this step may be repeated to refine the models until they are deemed appropriate, given the outputs of the previous step. The final result of this step should be one SID and a set of SBDs for each chosen process. Ideally, the models should be in a format that allows for standardized data exchange (e.g., the semantic exchange standard proposed in [35]). However, existing modeling and run-time environments from the commercial sector often make use of their own formats.
- Following model creation, the next step will be to set up a run-time environment for the subject-oriented process models and deploy them to it. If the previous steps already made use of such tools, then they can simply be re-used. A key requirement for the run-time environment is the ability to keep the running model instances synchronized with the state of the twinned process. This can be accomplished through a dedicated feature of the run-time environment itself, or through it offering the freedom of executing arbitrary code as part of a SBD state, allowing the implementation of features to request or receive status information of the twinned system to control instance execution (as sketched in Figure 9).
- The next step concerns establishing the infrastructure for receiving and storing the required information from the twinned process and the physical behavior carriers. The data also needs to be provided to future consumers (visualization). Based on the identified goals of the DT, the explored use cases, and the gathered requirements, the data from the system that is needed is determined. Once the required data is known, a decision needs to be made with regard to how it should be stored. The gen-

eral format for the event log was outlined in the previous sub-section and an example is shown as part of Figure 9, but, e.g., the type of database(s) to be used should consider the expected types of data and their other characteristics (e.g.,: volume, velocity). Access to the database(s) can be encapsulated through various services built on top of it.

- Next, the cyber-physical components that enact the modeled behavior need to provide the needed data to the storage infrastructure that was set up. This means that the components that need to provide data require some form of network connectivity. Furthermore, adaptations and extensions to their existing behavior will be necessary. Here, the impact that the additional communication would have on the performance of the components needs to be carefully considered. It might be likely that not all the data that was initially identified as potentially interesting for the twin can actually be provided without impacting the system in a significant manner. This is the reason, why these considerations were already mention at the beginning steps of the proposed methodology. Furthermore, the technologies used for realizing communication need to consider Quality-of-Service requirements that might exist. To summarize, one of the most important technical considerations during implementation relates to getting the relevant data from each component.
- Once the infrastructure for getting the required data is in place, a user interface needs to be constructed that makes use of this data to provide various visualization features to DT users. Alongside event data, the created models can provide process context information, as shown in Figure 9.
- Before the system can be put into active operation, it needs to be evaluated with regard to the various requirements that were established. This step requires incorporating the DT’s intended users and other involved stakeholders to ensure that it supports them as envisioned. Depending on the results of the evaluation, some of the previous steps might need to be repeated to further fine-tune the digital twin.
- Once evaluation has been passed successfully, the created twin passes into the phase of active operation and utilization.
- The last step that we outline concerns maintaining the created digital twin system. If the twinned process changes, adaptations to the models and twin will be necessary. Similarly, new and changed stakeholder requirements might come up that require, e.g., additional data from components and new visualization features.

The steps described above are again summarized in Figure 10. The next sub-section will outline some of the possible usages and benefits of the behavior-centered approach using the illustration case.

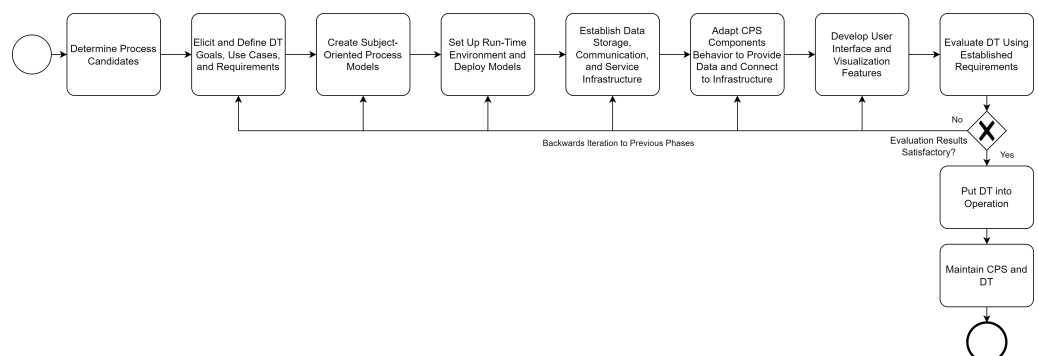


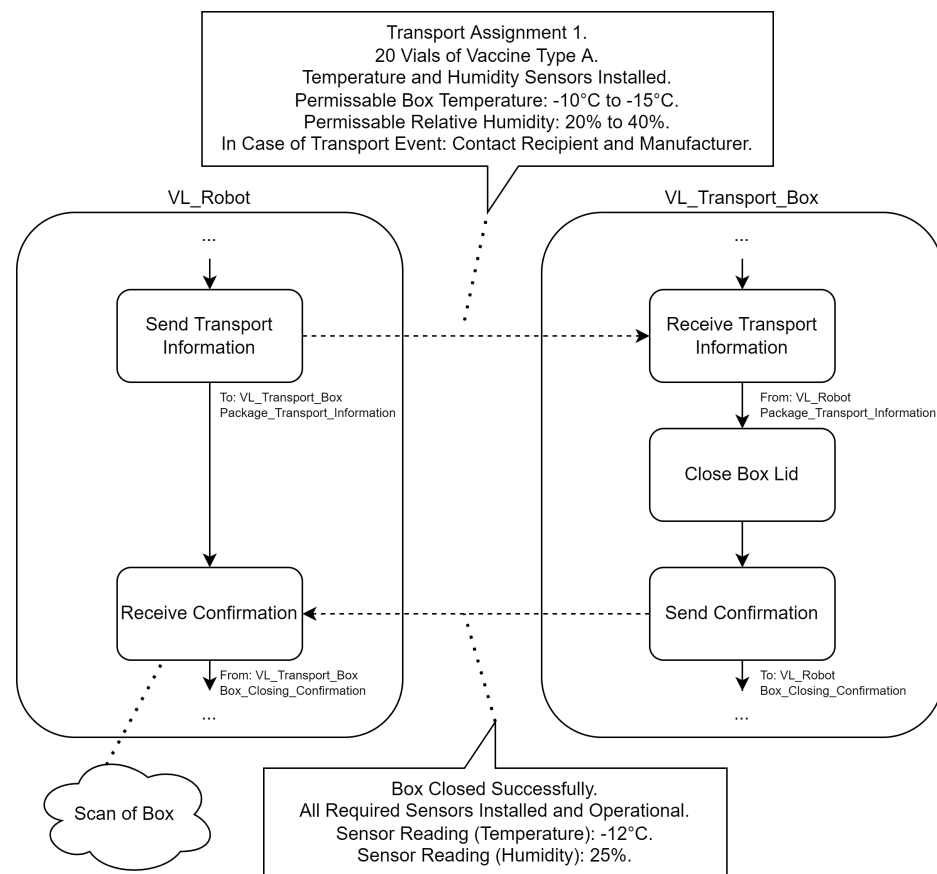
Figure 10. Generic steps for realizing the proposed digital twin architecture.

### 3.5. Exemplifying Use Case Scenarios

As outlined in the previous section, process model execution information is integrated with the (geo-spatial) context provided by the system at certain relevant points of the process. This supports the tracking of the logistics process and makes the system observable



for the stakeholders. Figure 11 illustrates this for the communication between the robot and the transport box upon successfully completing the outfitting of the box.



**Figure 11.** An example for the monitoring of the standard behavior of components and the related data. Data that concerns both a send state and a receive state was associated with the message connecting the two.

Different cross-cutting concerns with regard to the processes, the system, and the environment can be assessed by stakeholders. This goes beyond the monitoring of standard behavior. Considering the outlined vaccine use case, it would be possible to show the structural integrity of vaccine packaging (by, e.g., recognizing surface deformities) at different points in the process, to help rule out that damage occurred during transport preparation. The two following examples should help illustrate other possible applications of the proposed twin:

- **Digital Process Twin Application Example 1:** After the successful transportation of a box to its destination, the organization in charge of transport preparation receives the complaint that a sensor was not installed that should have been installed. The person in charge of supervising the process accesses past process model instances and looks at the one of the package in question. Both the model execution history and the scan results associated with states show that the sensor in question was installed properly, showcasing that it went missing after the package left the care of the organization. The behavioral twin of the box itself (enriched with the produced sensor readings) shows that it was operating as intended during the transport process as well, indicating that the sensor was probably removed at a later point in the process.
- **Digital Process Twin Application Example 2:** The person in charge of supervising the transport preparation process receives a notification on their smartphone from the robot that it cannot continue with its normal behavior due to unexpected changes in the environment that it cannot compensate for. The supervisor looks at the process

model instance that is being executed and sees that the robot is currently in the process of selecting a sensor for installation. Upon accessing and viewing the scan data associated with the current point in the process, it can be seen that the shelf was moved and sensors had fallen to the ground due to a minor earthquake or some other disturbance. Subsequently, the supervisor can react and restore the original environment based on what is known about the set-up (both a priori and from earlier scans as part of the process).

Instead of having to view the environmental data as a time-series of scan data without additional context, stakeholders can use the created process model instances (and the past model instance documentations stored) to immediately access the points in the process that they are interested in. The saved data can be used for both manual and automated analysis to gain insights regarding the CPS-supported business process and the CPS itself. These aspects of the proposed digital twin of the CPS are akin to process monitoring and auditing, as it is enabled through workflow execution engines in the context of business process management (the field from which subject-oriented modeling originates). Furthermore, considering the proposed DT model, one can opt to only create “snapshots” of the operational environment at crucial states in the process and store them, instead of continuously saving point clouds throughout the whole operation of the system, to minimize the data that needs to be stored.

#### 4. Related Work

In this section, we first examine existing research pertaining to geo-spatial information provided by LiDAR technology and its usage as part of a digital twin. The goal is to position and discuss our proposed approach and the used conception of digital twins in relation to how they are usually understood in the literature and different application contexts. We furthermore take a look at different methodologies that were proposed for the creation of digital twins, bringing our approach into the context of existing research and identifying potential shortcomings and opportunities for further research.

##### 4.1. Geo-Spatial Context, LiDAR Technology, and Digital Twins

A wide selection of existing research from different fields focuses on the geo-spatial information of some sort of asset itself (or of multiple assets) as the central building block of a digital twin.

In [34], LiDAR scan data was used to create a digital representation of a bridge that can be viewed in a virtual reality environment to facilitate bridge inspection. The authors in [36] proposed a framework for the geometric quality assessment of façades, based on digital twins. A 3D as-built digital replica (enriched with semantic information) is created from LiDAR point cloud data of the physical façade and compared to the as-designed model. In [37], a methodology for the creation of digital twins of large-scale structures based on RGB images and LiDAR scans was proposed and demonstrated for a wind turbine transition piece. The twin can be updated in regular intervals to reflect the current physical state of the structure. Specifically, based on the comparison between 3D reconstruction and the initial design of the structure, geometric deviations and production tolerances can be discovered and assessed. The authors also provided an algorithm for detection and classification of paint defects from the captured images, which can be mapped to the 3D reconstruction.

In [14], a workflow for creating digital twins of trees from 3D point clouds produced by a mobile LiDAR scanner was discussed. The authors of [38] aimed to automate the population of a digital twin of the city of Singapore with tree models (following the CityGML standard), with (mobile and airborne) LiDAR being among the used data sources. The modeling of city objects from LiDAR point clouds to subsequently support the creation of digital twin cities was discussed in [39], with the authors providing an unsupervised method.

In [13], an automated inspection system for power lines was described, using a multi-modal sensor system, including LiDAR technology. Sensor data and inspection results are

organized to create a digital twin of the power line. The 3D point cloud is semantically segmented, a 2D object detector is run on captured RGB camera images, and the resulting 2D bounding boxes are combined with the segmented point cloud. This forms the basis for the digital twin representation of the line. Similarly, in [16], a system for vegetation management in power line corridors was proposed. To this end, the authors created a digital twin by fusing data derived from LiDAR scans together with other relevant data sources through their proposed processing pipeline. They also provided approaches for encroaching vegetation detection and data segmentation for learning vegetation growth simulation.

In [40], a digital twin of an intersection was created to enable the real-time monitoring of intervisibility between pair-wise conflicting agents. To this end, point cloud data and trajectory data of traffic participants is used as inputs. First, conflict points and their associated agents are identified, and agents are modeled as 3D bounding cuboids, representing road users in real-time. Subsequently, dynamic line-of-sight analysis is performed and virtual warning signals can be given based on the results.

Compared to the digital twin conceptions underlying most of the approaches listed above, where the focus lies on the geo-spatial information itself as core of the twin, our approach starts with the behavior of certain active assets in an environment (CPS components) and adds the geo-spatial information as context (together with and in relation to other context data) to increase the semantic richness of the behavioral twin. For certain examined applications, the geo-spatial context information needed for generating the twin is captured once, e.g., to facilitate one inspection task of the twinned asset (e.g., [34,36]). The twin is sometimes also consistently updated to correctly reflect the state of the physical object(s) and changes over time (e.g., [13,37]), depending on the underlying use case. The life-cycle perspective supported by keeping the twin up-to-date and remembering previous data is also a crucial aspect of our proposed digital twin.

Approaches like [13] also aim to structure the information that is part of a digital twin in a form that facilitates the work of the primary stakeholders (in this case the power line operators). However, for many of the scenarios listed above, the behavior of the twinned object is not necessarily relevant for the tasks that its users should be supported in. Therefore, the focus lies on the geo-spatial information as anchor-point for integrating other data. For scenarios like the presented illustration case, where system behavior over time plays a crucial role, our digital twin architecture can provide benefit to its stakeholders by allowing access to information structured along a relevant business process. Here, the used subject-oriented notation can depict behavior of heterogeneous actors in a unifying way and with a minimal number of notational elements, facilitating the intelligibility of digital models for process stakeholders.

A selection of existing research also concerns the digital twinning of the LiDAR technology itself, either as a singular entity or as part of a larger system. Such a digital twin was, e.g., discussed in [41], with the authors aiming to build a reduced complexity model of a Flash-LiDAR system as basis. The intended DT usage concerns performing simulations and producing point cloud data, which can be used as basis for the training of neural networks that can subsequently perform detection and correction of aberrations in the LiDAR system. In [42], the authors used a digital twin to evaluate the potential of LiDAR mounted on an unmanned aerial system to detect various air collision risks, such as objects that are small, uncooperative, and unpredictable. A twin of the port of Hamburg was enriched with the necessary agents (air collision hazards, unmanned aerial system, mounted LiDAR simulation) to simulate and test different scenarios. Similarly, the authors of [43] created a digital twin to simulate and test a virtual service robot aimed at providing care to the elderly or people with development disabilities within its environment. The functionalities of the socially assistive robot that were tested included robot navigation, fall detection of a patient, and gesture recognition. The authors stated that three twins were created, one of the room, the service robot, and an elderly virtual avatar, respectively. As part of the robot twin, a virtual LiDAR sensor that generates data was simulated. In [44], a digital twin of a CPS was created. The twinned system encompasses two industrial

robots automatically conducting measurements for LiDAR sensors aimed to be used in the aerospace sector. For this purpose, one robot holds a LiDAR sensor and the other holds a target material. Digital twins of the different components were created, including the LiDAR sensor. Different twins were used in a variety of different ways, acting as a virtual test object (to plan trajectories), virtual mirror (to observe the current state), mediator (to actuate the real object), and mental model.

As part of our approach, the LiDAR sensor itself is insofar part of the digital twin, as that it can be either modeled as its own subject with the corresponding behavior or included within another subject representing the behavior of a larger CPS component (e.g., the robot system as a whole). The data provided by the sensor is given to the digital twin and documented alongside (current or past) instances of behavior. The focus therefore lies on run-time monitoring of the system. In comparison, most of the existing approaches outlined above focus on the simulation of a system to test different functionality. In [42,43], the LiDAR sensor is considered because the data that it provides is crucial for what should be simulated, tested, and evaluated. In [41], the twin aims to support data collection for the purpose of training neural networks for LiDAR recalibration. The role of digital twins as virtual mirrors described in [44] and demonstrated for a CPS corresponds more closely to the described usage of our digital twin architecture. However, none of these approaches make use of capturing and modeling system (component) behavior, and thus, operational processes in the vein of the proposed digital twin.

#### 4.2. Methodologies for the Creation of Digital Twins

Next, we review existing research contributions aimed towards establishing and applying methodologies for the creation of Digital Twins.

The authors in [45] conducted a review of articles that address the development of digital twins, specifically evaluating approaches from the domains of product life-cycle management, manufacturing, and predictive maintenance. They consequently synthesized three domain-specific development models, which were consolidated into a single model. The integrated model presented by the authors encompasses nine development steps: Prepare, Design, Verify/Validate, Deploy, Use, Evaluate, Maintain, Tune, and Rebuild.

The authors in [46] proposed a multi-level design methodology for the creation of digital twins, allowing for different levels of abstractions for the same models. The methodology takes an AutomationML (see <https://www.automationml.org/about-automationml/automationml/>, retrieved 18 October 2022) description of a production system as input and subsequently creates a model of the plant topology and a communication infrastructure to connect the used simulators (plant and process).

A general and domain-independent design methodology was proposed in [47]. The design starts with a purpose definition for the DT, which is followed by an identification of the process or asset that should be twinned, a DT usage specification, identification of fitting technologies, and the definition of input and output parameters. These steps are followed by development, performance measurement, and, based on the results of the measurement, deployment or the revisiting of design activities.

A more technology-centered approach, accompanying a corresponding reference architecture, was proposed in [48]. The authors outlined a design and deployment methodology using specifically AutomationML and web services. AutomationML is used to first model physical devices and then the DT itself. Subsequently, the parameters of the DT are configured and it is possible to repeat the previous steps for further refinement. Finally, the created models are used as input for deployment via a custom Python script.

The authors in [49] proposed an approach for the creation of digital models as a preparatory and enabling step for the subsequent creation of DTs. The specific types of models addressed are physics-based and the outlined methodology consists of three phases. First, the dynamic behavior of a machine is modeled. Then, virtual sensors are modeled next, which are used to obtain data from the model during simulation. Finally, the modeling parameters are defined that are required to adjust the model to reflect the real machine.

In [50], a development methodology for the DTs was proposed that was based on the V-model. This model originated in the field of Software Engineering and was later adapted and specified for mechatronic systems and CPS in the guideline VDI 2206 issued by the Association of German Engineers (see <https://www.vdi.de/richtlinien/programme-zu-vdi-richtlinien/vdi-2206/>, retrieved 18 October 2022). The authors outlined that their approach is intended for physical IoT-based products, with a focus on improving overall sustainability. They additionally considered four different initiation scenarios (simultaneous development of product, services, and DT; successive development of a DT for an existing product, with simultaneous upgrade of product to next version; ...) and allowed for multiple development cycles with different foci (product, DT, service), with each having its own V-model.

Overall, it was observed that DT design methodologies can be given at many different levels of abstraction and generality. There exist more abstract and general proposals that can be used to describe and structure design and development approaches from a variety of fields and concerning a variety of different flavors of DTs and real-world entities to be twinned (e.g., [45,47] go in this direction). In general, rough correspondences between the proposed steps of the examined approaches and the ones of our methodology can be identified (e.g., initial steps of purpose definition and selection of assets/process to be twinned in [47], verification and validation steps in [45,50], iterations for adaption/refinement of the DT in [45,47,48]). However, some of the parts of these models (e.g., the last four sub-processes in [45], or the various development steps of the DT V-model in [50]) have not been covered by the proposed development approach to a comparable level of detail.

Then, there also exist specific approaches for a certain type of asset (e.g., manufacturing systems [46]), or a certain type of twin (e.g., in terms of content of the twin, such as physics-based models [49]), or even a specific part of the wider DT creation process (e.g., model creation [49]). In relation to the existing research, the introduced methodology for DT design can be also considered as a specific instantiation for a certain class of twin (behavior-centered process twins), with cyber-physical assets being the main real-world entities of interest. The focus on processes from more of a business context is a unique aspect of the introduced methodology (the term “process” is also used by some of the examined works with a different connotation, e.g., in [46] it refers to physical (kinematic) processes).

The examination also revealed some of the limits and opportunities for the extension of our proposed methodology. By examining the identified steps of other proposals, certain aspects of design and development were uncovered that were not considered yet. For instance, certain phases related to the continuing CPS and DT life-cycle could be covered in more detail. The different variations for DT development scenarios listed in [50] are particularly interesting, since our proposal for now only covers the case of an existing CPS as entry point for DT creation. The approach in [48] furthermore supports its users through some degree of automation: After the sufficient refinement of models, automatic deployment of the DT can be accomplished. Such support could greatly ease development and adoption efforts. Creating an expanded methodology by taking into consideration and integrating aspects of other approaches is considered progressing the presented research.

In this section, we have reviewed some of the related work pertaining to existing DT approaches that use geo-spatial context provided by LiDAR technology and methodologies for the creation of DTs. The focus on approaches outside the field of subject orientation reveals the need for behavior-centered and integrated DT design and engineering. Hence, as outlined in Section 3.1, we utilized existing behavior-centered modeling and execution as basis for extending the knowledge base of subject orientation, and applied this behavior-centered approach for geo-spatial sensitive CPS and DTs, as discussed in the following section.

## 5. Discussion

In the previous section, we reviewed relevant related work pertaining to existing DT approaches that use geo-spatial context provided by LiDAR technology and methodologies



for the creation of DTs. We discussed our research contribution in light of the existing approaches. Taking into concern DT-related approaches not considering process twins and subject orientation for geo-spatial CPS development so far, the presented approach extends the existing knowledge base of subject orientation and the application of this behavior-centered approach for this type of CPS and DTs (which were outlined in Section 3.1).

Specifically, our approach was built on the digital twin conception presented in [8]. The authors of this article focused on designing a subject-oriented DT as a means of developing a CPS through behavior specification. They detailed how to depict certain aspects of CPS structure and behavior (such as variability of behavior) through subject orientation and associated modeling techniques and exemplified this through evolving models for a traffic management scenario. Data exchange between the DT and the CPS were established as a core aspect of the concept, without going further into details on how this data can be structured. With our contribution in this article, we detailed the initial concept of a behavior-centered DT in terms of how this integration between the data of a CPS and the process models at the core of this DT-type can look like and work effectively in the course of system design and engineering. We described this integration specifically centered around geo-spatial context data, a type of data with considerable importance in the CPS context, as was outlined in the introduction. With the architecture depicted in Figure 9 and the guiding methodology steps in Figure 10, including the associated considerations, we specified how an infrastructure set-up for this type of twin can be realized. We also discussed some of the modeling-related aspects and provided sample model variants for the illustration case, in the respect that they are relevant to the proposed architecture and methodology.

As was also identified in the previous section, a point that is still open for exploration concerns the automatic deployment of subject-oriented models. This is also meant in the sense of a run-time environment ensuring automatic execution of models through the CPS and its components, as well as them providing the relevant data to the environment without further configuration and engineering activities (as they are currently required in the proposed methodology). Furthermore, this also represents a core step towards the idea of design-integrated engineering presented in [8], where the CPS is developed and adapted dynamically through the models in the vein of Model-Driven Engineering and based on the idea of the DT as a virtual entity connected with the twinned entity in such a way that changes to one are reflected in the other.

## 6. Conclusive Summary

LiDAR technology has the potential to provide valuable information for the operation of Cyber-Physical Systems. This information is considered useful for stakeholders in charge of monitoring the system as it supports and realizes related business processes. A big part of the vision and potential of digital twins lies in linking relevant information from multiple sources to support different tasks and users. A digital twin of a CPS should therefore integrate and provide geo-spatial context information in a form that facilitates stakeholder accessibility.

In this article, we showed the potential of a behavior-centered digital twin conception that utilized subject-oriented process models to structure the data provided by a CPS. We referred to concepts from process monitoring to document the past behavior of a system (and its components), including the associated context information. We demonstrated that geo-spatial information can be integrated with other data provided by the CPS to create a detailed snapshot of the system and its operational environment, which can be associated with a concrete behavior state of a CPS component. We proposed a corresponding architecture concept and methodology guiding realization.

Utilizing an illustration case from the logistics domain, we explored some of the benefits such type of a twin brings to stakeholders in charge of monitoring processes and the CPS. Stored information can be accessed directly from the relevant points in the process, ideally supported through a visual representation of both process instances and the



associated context information. The logged data can also be used as input for automated inspection. Finally, we examined related work concerning the combination of digital twins and geo-spatial information provided by LiDAR technology. We could position our approach as not only advancing model-based engineering towards behavior-driven development, but providing it up-front as the primary core of the twin.

Most of the existing research focuses on using the LiDAR-generated data to create a digital representation of an asset (sometimes through integration with other relevant data). For scenarios, where CPS are used to support and help realize business processes, the behavior of active agents is of crucial importance compared to monitoring a mostly “static” asset. It is for such cases that our DT conception offers contextual knowledge and stakeholder value. To that respect, the body of knowledge of subject orientation in the areas of CPS and behavior-centered DTs was enriched.

Through examining existing methodologies for the creation of DTs, we also identified some shortcomings of our research in terms of the supported development scenario and automation support of methodological steps. A limitation of this work is that the proposed DT architecture, as presented conceptually, requires future work on implementation. Its efficacy for operational CPS scenarios needs to be evaluated involving process stakeholders from relevant application domains to provide empirical evidence. Potential for future research also exists with regard to the visualization of process model instances and the connected geo-spatial information. Extensions to subject-oriented run-time engines for this purpose are required to ensure a suitable presentation of DT information. Finally, support for automating the deployment of models in cyber-physical settings and establishing the required data exchange through subject-oriented run-time environments is a desirable feature to ease the adoption of the outlined concept. The adaption of automated process monitoring and mining techniques to the presented CPS context is also of interest to optimize and further develop systems and processes based on DT data.

**Author Contributions:** Conceptualization, T.E.J., C.S. and R.H.; methodology, T.E.J.; software, T.E.J.; validation, T.E.J., C.S. and R.H.; formal analysis, T.E.J.; investigation, T.E.J., C.S. and R.H.; resources, T.E.J., C.S. and R.H.; data curation, T.E.J. and R.H.; writing—original draft preparation, T.E.J. and C.S.; writing—review and editing, T.E.J., C.S. and R.H.; visualization, T.E.J.; supervision, T.E.J. and C.S.; project administration, T.E.J. and R.H. All authors have read and agreed to the published version of the manuscript.

**Funding:** Published with the support of the Johannes Kepler University’s Publication Fund.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

AGV	Automated Guided Vehicle
BPMN	Business Process Model and Notation
CPS	Cyber-Physical System
DT	Digital Twin
LiDAR	Light Detection and Ranging
SBD	Subject Behavior Diagram
S-BPM	Subject-Oriented Business Process Management
SID	Subject Interaction Diagram
SLAM	Simultaneous Localization and Mapping
UML	Unified Modeling Language

## References

1. Jazdi, N. Cyber physical systems in the context of industry 4.0. In Proceedings of the 2014 IEEE International Conference on Automation, Quality and Testing, Robotics, Cluj-Napoca, Romania, 22–24 May 2014; pp. 1–4. [CrossRef]
2. Roriz, R.; Cabral, J.; Gomes, T. Automotive LiDAR Technology: A Survey. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 6282–6297. [CrossRef]
3. Ismail, H.; Roy, R.; Sheu, L.J.; Chieng, W.H.; Tang, L.C. Exploration-Based SLAM (e-SLAM) for the Indoor Mobile Robot Using Lidar. *Sensors* **2022**, *22*, 1689. [CrossRef] [PubMed]
4. Söderberg, R.; Wärmefjord, K.; Carlson, J.S.; Lindkvist, L. Toward a Digital Twin for real-time geometry assurance in individualized production. *CIRP Ann.* **2017**, *66*, 137–140. [CrossRef]
5. Shafto, M.; Conroy, M.; Doyle, R.; Glaessgen, E.; Kemp, C.; LeMoigne, J.; Wang, L. Draft modeling, simulation, information technology & processing roadmap. *Technol. Area* **2010**, *11*, 1–27. Available online: [https://www.nasa.gov/pdf/501321main\\_TA11-MSITP-DRAFT-Nov2010-A1.pdf](https://www.nasa.gov/pdf/501321main_TA11-MSITP-DRAFT-Nov2010-A1.pdf) (accessed on 20 September 2022).
6. Tao, F.; Zhang, H.; Liu, A.; Nee, A.Y.C. Digital twin in industry: State-of-the-art. *IEEE Trans. Ind. Inf.* **2019**, *15*, 2405–2415. [CrossRef]
7. Madni, A.M.; Madni, C.C.; Lucero, S.D. Leveraging digital twin technology in model-based systems engineering. *Systems* **2019**, *7*, 7. [CrossRef]
8. Stary, C.; Elstermann, M.; Fleischmann, A.; Schmidt, W. Behavior-Centered Digital-Twin Design for Dynamic Cyber-Physical System Development. *Complex Syst. Inf. Model. Q.* **2022**, *30*, 31–52. [CrossRef]
9. Kannengiesser, U.; Frysak, J.; Stary, C.; Krenn, F.; Müller, H. Developing an engineering tool for cyber-physical production systems. *E I Elektrotechnik Inf.* **2021**, *138*, 330–340. [CrossRef]
10. Autiosalo, J.; Vepsäläinen, J.; Viitala, R.; Tammi, K. A Feature-Based Framework for Structuring Industrial Digital Twins. *IEEE Access* **2020**, *8*, 1193–1208. [CrossRef]
11. Julien, N.; Martin, E. How to characterize a Digital Twin: A Usage-Driven Classification. *IFAC-PapersOnLine* **2021**, *54*, 894–899. [CrossRef]
12. Zhang, J.; Zhao, X.; Chen, Z.; Lu, Z. A Review of Deep Learning-Based Semantic Segmentation for Point Cloud. *IEEE Access* **2019**, *7*, 179118–179133. [CrossRef]
13. Kähler, O.; Hochstöger, S.; Kemper, G.; Birchbauer, J. Automating powerline inspection: A novel multisensor system for data analysis using deep learning. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2020**, *XLIII-B4-2020*, 747–754. [CrossRef]
14. Niță, M.D. Testing Forestry Digital Twinning Workflow Based on Mobile LiDAR Scanner and AI Platform. *Forests* **2021**, *12*, 1576. [CrossRef]
15. Clark, T.; Brock, E.; Wu, D.; Liang, Y. Development of Real-Time Smart City Mapping Utilizing Game Engines. In Proceedings of the 2020 International Symposium on Networks, Computers and Communications (ISNCC), Montreal, QC, Canada, 20–22 October 2020; pp. 1–6. [CrossRef]
16. Mongus, D.; Brumen, M.; Žlaus, D.; Kohek, Š.; Tomažič, R.; Kerin, U.; Kolmanič, S. A Complete Environmental Intelligence System for LiDAR-Based Vegetation Management in Power-Line Corridors. *Remote Sens.* **2021**, *13*, 5159. [CrossRef]
17. Margaria, T.; Schieweck, A. The digital thread in industry 4.0. In *Proceedings of the International Conference on Integrated Formal Methods, Bergen, Norway, 2–6 December 2019*; Springer: Cham, Switzerland, 2019; pp. 3–24. [CrossRef]
18. Conde, J.; Munoz-Arcentales, A.; Alonso, A.; Lopez-Pernas, S.; Salvachua, J. Modeling digital twin data and architecture: A building guide with fiware as enabling technology. *IEEE Internet Comput.* **2021**, *26*, 7–14. [CrossRef]
19. Shirowzhan, S.; Tan, W.; Sepasgozar, S.M. Digital twin and CyberGIS for improving connectivity and measuring the impact of infrastructure construction planning in smart cities. *ISPRS Int. J. Geo-Inf.* **2020**, *9*, 240. [CrossRef]
20. Moshrefzadeh, M.; Machl, T.; Gackstetter, D.; Donaubaue, A.; Kolbe, T.H. Towards a distributed digital twin of the agricultural landscape. *J. Digit. Landsc. Archit.* **2020**, *5*, 173–186. [CrossRef]
21. Zhang, H.; Cheng, S.; Niu, L.; Clark, A. Barrier Certificate based Safe Control for LiDAR-based Systems under Sensor Faults and Attacks. *arXiv* **2022**, arXiv:2208.05944.
22. Wang, J.; Li, X.; Wang, P.; Liu, Q. Bibliometric analysis of digital twin literature: A review of influencing factors and conceptual structure. *Technol. Anal. Strateg. Manag.* **2022**, 1–15. [CrossRef]
23. Fleischmann, A.; Schmidt, W.; Stary, C.; Obermeier, S.; Börger, E. *Subject-Oriented Business Process Management*; Springer: Berlin/Heidelberg, Germany, 2012. [CrossRef]
24. Qi, C.R.; Su, H.; Kaichun, M.; Guibas, L.J. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 77–85. [CrossRef]
25. Qi, C.R.; Yi, L.; Su, H.; Guibas, L.J. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. In *Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS'17, Long Beach, CA, USA, 4–9 December 2017*; Curran Associates Inc.: Red Hook, NY, USA, 2017; pp. 5105–5114. [CrossRef]
26. Zhou, Y.; Tuzel, O. VoxelNet: End-to-End Learning for Point Cloud Based 3D Object Detection. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 4490–4499. [CrossRef]

27. Shi, S.; Wang, Z.; Shi, J.; Wang, X.; Li, H. From Points to Parts: 3D Object Detection From Point Cloud With Part-Aware and Part-Aggregation Network. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *43*, 2647–2664. [[CrossRef](#)]
28. Weichhart, G.; Reiser, M.; Stary, C. Task-Based Design of Cyber-Physical Systems—Meeting Representational Requirements with S-BPM. In *Subject-Oriented Business Process Management. The Digital Workplace—Nucleus of Transformation*; Freitag, M., Kinra, A., Kotzab, H., Kreowski, H.J., Thoben, K.D., Eds.; Springer International Publishing: Cham, Switzerland, 2020; Volume 1278, pp. 63–73. [[CrossRef](#)]
29. Moattar, H.; Bandara, W.; Kannengiesser, U.; Rosemann, M. Control flow versus communication: Comparing two approaches to process modelling. *Bus. Process Manag. J.* **2022**, *28*, 372–397. [[CrossRef](#)]
30. Elstermann, M.; Ovtcharova, J. Sisi in the ALPS: A Simple Simulation and Verification Approach for PASS. In *Proceedings of the 10th International Conference on Subject-Oriented Business Process Management—S-BPM One '18, Linz, Austria, 5–6 April 2018*; ACM Press: Linz, Austria, 2018; pp. 1–9. [[CrossRef](#)]
31. Krenn, F.; Stary, C. Exploring the Potential of Dynamic Perspective Taking on Business Processes. *Complex Syst. Informatics Model. Q.* **2016**, *8*, 15–27. [[CrossRef](#)]
32. Verenich, I.; Dumas, M.; Rosa, M.L.; Maggi, F.M.; Teinemaa, I. Survey and Cross-benchmark Comparison of Remaining Time Prediction Methods in Business Process Monitoring. *ACM Trans. Intell. Syst. Technol.* **2019**, *10*, 34. [[CrossRef](#)]
33. Cao, R.; Ni, W.; Zeng, Q.; Lu, F.; Liu, C.; Duan, H. Remaining time prediction for business processes with concurrency based on log representation. *China Commun.* **2021**, *18*, 76–91. [[CrossRef](#)]
34. Omer, M.; Margetts, L.; Hadi Mosleh, M.; Hewitt, S.; Parwaiz, M. Use of gaming technology to bring bridge inspection to the office. *Struct. Infrastruct. Eng.* **2019**, *15*, 1292–1307. [[CrossRef](#)]
35. Elstermann, M.; Krenn, F. The Semantic Exchange Standard for Subject-Oriented Process Models. In *Proceedings of the 10th International Conference on Subject-Oriented Business Process Management—S-BPM One '18, Linz, Austria, 5–6 April 2018*; ACM Press: Linz, Austria, 2018; pp. 1–8. [[CrossRef](#)]
36. Tran, H.; Nguyen, T.N.; Christopher, P.; Bui, D.K.; Khoshelham, K.; Ngo, T.D. A digital twin approach for geometric quality assessment of as-built prefabricated façades. *J. Build. Eng.* **2021**, *41*, 102377. [[CrossRef](#)]
37. Benzon, H.H.; Chen, X.; Belcher, L.; Castro, O.; Branner, K.; Smit, J. An Operational Image-Based Digital Twin for Large-Scale Structures. *Appl. Sci.* **2022**, *12*, 3216. [[CrossRef](#)]
38. Gobeawan, L.; Lin, E.S.; Tandon, A.; Yee, A.T.K.; Khoo, V.H.S.; Teo, S.N.; Yi, S.; Lim, C.W.; Wong, S.T.; Wise, D.J.; et al. Modeling trees for virtual singapore: From data acquisition to citygml models. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2018**, *XLII-4/W10*, 55–62. [[CrossRef](#)]
39. Xue, F.; Lu, W.; Chen, Z.; Webster, C.J. From LiDAR point cloud towards digital twin city: Clustering city objects based on Gestalt principles. *ISPRS J. Photogramm. Remote Sens.* **2020**, *167*, 418–431. [[CrossRef](#)]
40. Ma, Y.; Zheng, Y.; Wong, Y.D.; Easa, S.; Cheng, J. A virtual procedure for real-time monitoring of intervisibility between conflicting agents at intersections using point cloud and trajectory data. *Transp. Res. Part C Emerg. Technol.* **2022**, *134*, 103486. [[CrossRef](#)]
41. Tavakolibasti, M.; Meszmer, P.; Bottger, G.; Kettelgerdes, M.; Elger, G.; Erdogan, H.; Seshaditya, A.; Wunderle, B. Thermo-mechanical-optical coupling within a digital twin development for automotive LiDAR. In *Proceedings of the 2021 22nd International Conference on Thermal, Mechanical and Multi-Physics Simulation and Experiments in Microelectronics and Microsystems (EuroSimE)*, St. Julian, Malta, 19 April 2021; pp. 1–9. [[CrossRef](#)]
42. Riordan, J.; Manduhu, M.; Black, J.; Dow, A.; Dooly, G.; Matalonga, S. LiDAR Simulation for Performance Evaluation of UAS Detect and Avoid. In *Proceedings of the 2021 International Conference on Unmanned Aircraft Systems (ICUAS)*, Athens, Greece, 15–18 June 2021; pp. 1355–1363. [[CrossRef](#)]
43. Alves, S.F.R.; Uribe-Quevedo, A.; Chen, D.; Morris, J.; Radmard, S. Developing a VR Simulator for Robotics Navigation and Human Robot Interactions employing Digital Twins. In *Proceedings of the 2022 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, Christchurch, New Zealand, 12–16 March 2022; pp. 121–125. [[CrossRef](#)]
44. Dahmen, U.; Priggemeyer, M.; Rossmann, J. Cyber-Physical Systems and Digital Twins in Practice – A Real-Life Application Example. In *Proceedings of the 2021 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE)*, Brisbane, Australia, 8–10 December 2021; pp. 1–8. [[CrossRef](#)]
45. Heindl, W.; Stary, C. Structured Development of Digital Twins—A Cross-Domain Analysis towards a Unified Approach. *Processes* **2022**, *10*, 1490. [[CrossRef](#)]
46. Centomo, S.; Avogaro, A.; Panato, M.; Tadiello, C.; Fummi, F. A Design Methodology of Multi-level Digital Twins. In *Proceedings of the 2021 22nd IEEE International Conference on Industrial Technology (ICIT)*, Valencia, Spain, 10–12 March 2021; pp. 961–966. [[CrossRef](#)]
47. Psarommatis, F.; May, G. A literature review and design methodology for digital twins in the era of zero defect manufacturing. *Int. J. Prod. Res.* **2022**, 1–21. [[CrossRef](#)]
48. Schroeder, G.N.; Steinmetz, C.; Rodrigues, R.N.; Henriques, R.V.B.; Rettberg, A.; Pereira, C.E. A Methodology for Digital Twin Modeling and Deployment for Industry 4.0. *Proc. IEEE* **2021**, *109*, 556–567. [[CrossRef](#)]
49. Aivaliotis, P.; Georgoulas, K.; Arkouli, Z.; Makris, S. Methodology for enabling Digital Twin using advanced physics-based modelling in predictive maintenance. *Procedia CIRP* **2019**, *81*, 417–422. [[CrossRef](#)]
50. Riedelsheimer, T.; Gogineni, S.; Stark, R. Methodology to develop Digital Twins for energy efficient customizable IoT-Products. *Procedia CIRP* **2021**, *98*, 258–263. [[CrossRef](#)]