http://dx.doi.org/10.25673/116039

„Ulrich Jumar, Christian Diedrich (Hrsg.):
EKA 2024 - Entwurf komplexer Automatisierungssysteme, 18. Fachtagung“

Methods to Describe, Assign and Derive Capabilities from the Capability, Skill and Service (CSS) Model

Themengebiet: Modellierung mit formalen Beschreibungen

Michael Winter[1], Jürgen Bock[2], Johannes Hoos[3], Tobias Klausmann[4], Boris Schnebel[5], Siwara Schmitt[6], Marco Simon[7], Melanie Stolze[8], Luis Miguel Vieira da Silva[9] und Tobias Kleinert[10]

Abstract: For the use of the CSS model, workflows are needed to describe, assign and derive capabilities. This contribution provides a Capability Determining method that encompasses the three underlying technology-independent methods to guide stakeholders in describing, assigning and deriving capabilities for manufacturing. The first underlying method, Describing Capabilities, describes a workflow for the formal semantic description of capabilities. The workflow for assigning capabilities to a production resource is explained using the second method Assigning Capability. In the last workflow, capabilities are derived from a production specification or a process description using the method Deriving Capability.

Keywords: Industry 4.0, Capability description, Resource function, Product description, Automation engineering

[1] RWTH Aachen University, Chair of Information and Automation Systems for Process and Material Technology (IAT), Turmstraße 46, 52064 Aachen, m.winter@iat.rwth-aachen.de
[2] Technische Hochschule Ingolstadt, AI-Applications in innovative Production and Logistics, Esplanade 10, 85049 Ingolstadt, juergen.bock@thi.de
[3] Festo SE & Co. KG, Advanced Develop. Digital Engineering, Ruiter Straße 82, 73734 Esslingen, johannes.hoos@fest.com
[4] Lenze SE, R&D Systems Engineering, Hans-Lenze-Straße 1, 31855 Aerzen, tobias.klausmann@lenze.com
[5] Fraunhofer IOSB, Optronik, Systemtechnik und Bildauswertung, Fraunhoferstr. 1, 76131 Karlsruhe, boris.schnebel@iosb-extern.fraunhofer.de
[6] Fraunhofer IESE, Embedded Systems, Fraunhofer-Platz 1, 67663 Kaiserslautern, siwara.schmitt@iese.fraunhofer.de
[7] Technologie-Initiative SmartFactory KL e.V., Trippstadter Straße 122, 67663 Kaiserslautern, marco.simon@smartfactory.de
[8] ifak - Institut für Automation und Kommunikation e.V., Werner-Heisenberg-Str. 1, 39106 Magdeburg, melanie.stolze@ifak.eu
[9] Helmut Schmidt University, Institute of Automation Technology, Holstenhofweg 85, 22043 Hamburg, miguel.vieira@hsu-hh.de
[10] RWTH Aachen University, Chair of Information and Automation Systems for Process and Material Technology (IAT), Turmstraße 46, 52064 Aachen, kleinert@iat.rwth-aachen.de

1 Introduction

In the rapidly evolving and volatile environment of manufacturing, flexibility plays a central role to be successful. The term flexibility encompasses the incorporation of more product variants through one production facility, the ability to accommodate smaller batch sizes of different product variants and the reorganization of production orders with a reduction in associated efforts. To fulfill these requirements for flexibility, the authors are applying the Capability, Skill, and Service (CSS) model for the implementation of capability-based engineering. Extended from the product, process and resource (PPR) paradigm, the CSS model decouples product design from production engineering [Kö23]. A capability is considered the implementation-independent specification of a function, while a skill is defined as the implemented function specified by a capability, cf. [Di22]. In order to produce products by applying a respective production procedure, capabilities are required to realize the individual steps of the procedure. These required capabilities, in turn, have to be matched to the capabilities offered by available production resources [Di22]. To determine capabilities for product fabrication and ultimately to find a suitable sequence of feasible production steps, a matching between required capabilities and those offered by the resources is performed. In the current state of research, there are no technology-agnostic and multi-purpose methodologies for the formal description of capabilities to achieve this matching. Therefore, this contribution by the Plattform Industrie 4.0 working group "Semantic and Interaction of I4.0 Components" addresses the following research questions:

- How to methodically approach the formal description of capabilities?
- How to methodically approach the assignment of capabilities to resources?
- How to methodically derive capabilities from product or process descriptions?

The objective is to provide three technology-independent methods to guide stakeholders through describing, assigning and deriving capabilities for manufacturing.

The remainder of this paper begins with an overview of related publications in Section 2. In Section 3 the methods for the formal description of capabilities are presented. Finally, Section 4 concludes the presented methods and outlines future research.

2 Related Work

In [Fr22], an overview of capability and skill-based manufacturing systems is given by conducting a literature review. Research in the area of capabilities and skills is justified by the need for adaptability and flexibility in manufacturing and the associated process planning. In addition to a formal description of capabilities and skills, the property of being suitably designed for matching of requirements and functions of production resources – ultimately for process planning – was identified as one of the most common requirements for capabilities and skills. It is shown that different modeling methods, e.g., ontologies or

UML, and different technologies, e.g., AutomationML or OWL, are used to model capabilities. However, modeling issues such as inconsistent modeled capabilities or different levels of abstraction are named as the most common challenges for the use of capabilities and skills in manufacturing. Therefore, there is a need for a method to formally describe capabilities that is ideally technology-independent and thus can be used for different capability-based approaches. [Fr22]

A number of contributions to methodical modeling of capabilities use ontologies: The authors of [Jä19] present a systematic development process of an ontology to describe capabilities of manufacturing resources. The presented method is based on the ontology engineering methodology of [Su09] and consists of five phases: feasibility study, kickoff, refinement, evaluation, and application and evolution. The main focus of this method is to model general concepts of a capability model to represent the domain under consideration, and it only considers modeling with ontologies. Actual capability instances are then modeled in the evaluation phase, but no methodological approach is presented. Furthermore, only the view of a resource and its provided capabilities is described, so that e.g. the derivation of capabilities from product specifications or process descriptions is not considered. In [Kö20] a method is presented that automatically generates the sub-aspects machine structure, capabilities and skills of a capability model from different artifacts in order to generate the entire capability model. In particular, the machine structure and skills aspects can be created automatically. The capability aspect is created semi-automatically by using a modelling tool for capabilities based on the VDI 3682 standard for formalized process description and then automatically transferring the created JSON file to the ontology. Accordingly, this approach allows the creation of a specific capability model in an ontology with a specific tool, without considering other standards for processes or other approaches to capability modeling, such as resource or product-based ones.

Other approaches use the Asset Administration Shell (AAS) to describe capabilities of assets: In [HDM21], a graphical modeling framework is presented for modeling systems according to the AAS standard. The framework enables the modeling of AAS without requiring an in-depth understanding of AAS. It also allows the modeling of capabilities or processes using BPMN. However, no method is presented on how to specifically model capabilities, and no standards are used to describe or derive capabilities. The authors of [Gr23] emphasize the importance of engineering processes and the embedding of engineering information from numerous standards in such capability models or specifically in digital twins (like the AAS) for continuous engineering. Important general suggestions for information modeling are given, such as the integration of standards into the AAS, reusable and extensible information modeling, but also the updating of information across all phases of engineering. However, there is no specific methodological proposal regarding capabilities and which standards should be used from PPR perspective to model them.

3 Methods for Determining Capabilities

The Capability Determining method proposed in this contribution entails the multiple use of underlying methods for describing, assigning and deriving capabilities. The user can decide whether a capability shall be described formally, assigned to a production resource, derived from a product specification, or be derived from a process description, see Fig. 1.

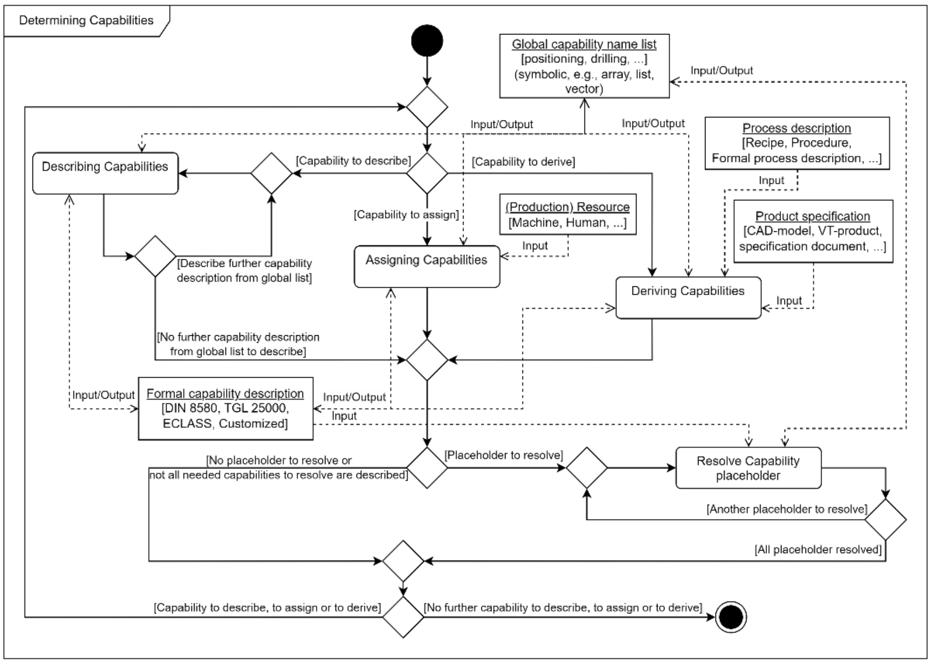

Fig. 1: Workflow for the method of Determining Capabilities

In the Capability Determining method various artifacts are used as inputs or outputs for the underlying methods. The main artifact that is used as an input and output for the underlying three methods Describing, Assigning and Deriving Capabilities is the formal capability description. This artifact can be, e.g., a capability submodel of the Asset Administration Shell or an ontology, which can consider several standards like DIN 8580, TGL 25000, ECLASS or customized function descriptions. Another artifact that serves as input and output for the underlying methods is the global capability name list. In this name list all formally described capabilities are available as key value pair, where the key is the capabilities unique id and the value is the capabilities informal name, e.g., “Drilling”. Depending on the underlying methods the name list might contain capability placeholders, where the placeholders are a temporary addendum. The capabilities for these placeholders must be described in the method Describing Capabilities and the placeholders must be resolved.

For a comprehensive understanding of the workflow regarding the description of a capability, the underlying method Describing Capabilities is used to formally describe capabilities, see Section 3.1.
The method Assigning Capabilities is used for assigning formally described capabilities to a resource to be used in capability-based production approaches, see Section 3.2. For this method a production resource, e.g., a machine or human, is used as input.
The method Deriving Capabilities allows the derivation of capabilities from product specifications or process descriptions for a possible automated match with assigned capabilities of resources, see Section 3.3.

In the case that no suitable capability description was found in the methods Assigning Capabilities or Deriving Capabilities, a capability placeholder (with a unique id) will be created. This placeholder is used as an entry in the global capability name list. The placeholder, which represents the capability to be described, is then formalized in the method Describing Capabilities. After the capability description is formalized with another unique id, the activity Resolve Capability placeholder is to apply. As the formal capability description might differ from the preliminary capability placeholder, the placeholder must be exchanged with the final unique id of the formal capability previously described. This activity can be repeated until all capability placeholders are resolved.

3.1 Describing Capabilities

The method Describing Capabilities is one of the underlying methods of the Capability Determining method. As shown in Fig. 2, the workflow of the method Describing Capabilities includes the general activities for a formal and semantic description of a capability.

The description of a capability specifies the context in which the capability will be applied. Depending on the context, capabilities can differ in their descriptions and properties, e.g., “grasping” a tomato (food industry) or “grasping” a component (manufacturing industry). The context can be refined by analyzing a production process or by comparing similar production processes, as well as by interviewing stakeholders to identify experiences and challenges.
Existing capability descriptions should be considered to avoid the creation of duplicates. Therefore, the introduced global capability name is to check whether the desired capability name already exists. Subsequently, if the capability name is identified within the list, the referenced formal capability description should be checked to see whether it matches the capability to be created. If the capability name is not found in the global capability name list, a thorough examination of other capability descriptions within the list is warranted to ascertain whether they encapsulate the desired capability but are listed under a different (synonymous) name. In this case, the capability name used or the capability name in the global list should be reconsidered. If no correspondence is discerned within the global

capability name list for the queried capability, alternative databases such as triple stores, Asset Administration Shell repositories, or publicly available sources including ontologies, ECLASS, or the IEC Common Data Dictionary (CDD) should be explored for further investigation.

If one of the identified existing capability descriptions suits well to the capability needed to be described, the workflow of describing capabilities is finished. For this, an existing capability description is evaluated to be sufficient by checking corresponding activities of the method Describing Capabilities and with a successful validation, the capability is resolved in the superior method Determining Capabilities to the global capability name list.

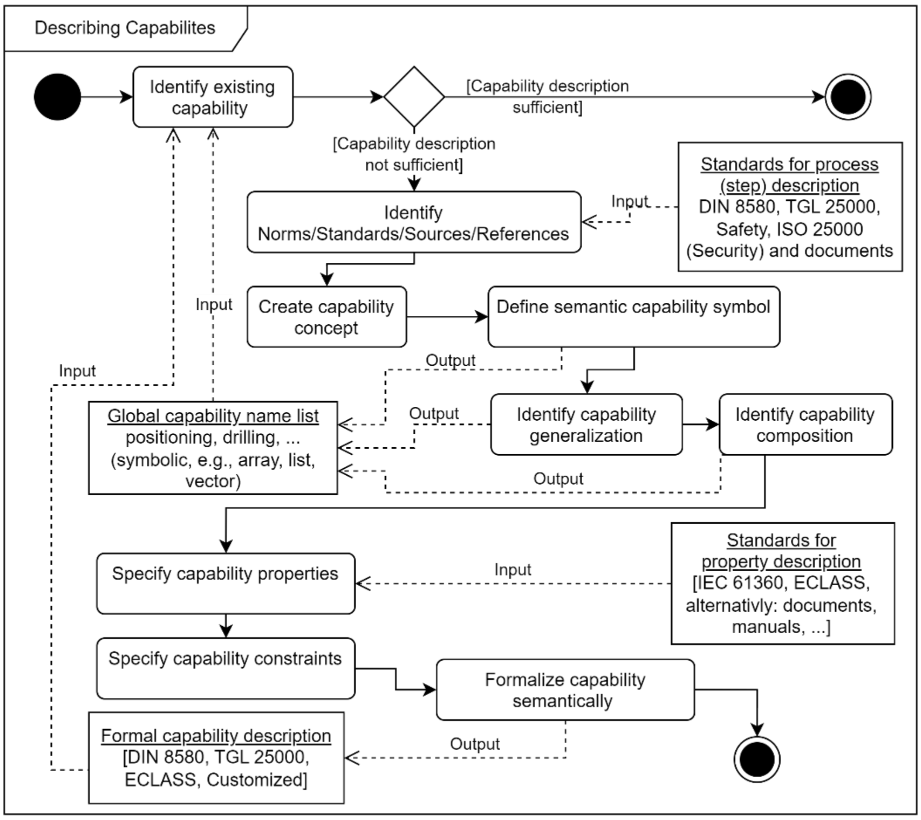

Fig. 2: Workflow for the method of Describing Capabilities

If no suitable existing capability is found, the next activity is to identify norms, standards, sources, or references that may contain information for the description of the capability. In the field of engineering and manufacturing, compliance with established norms, standards and references is crucial to ensure consistent quality and reliability of capabilities and processes. One such recognized standard is the DIN 8580, which defines the terminology for manufacturing processes. In process industry, the TGL 25000 is a standard that provides description of unit operations by two distinct classification schemas. Furthermore, safety and security information can be essential. This can include ISO 25000 for defining system and software quality requirements and their evaluation, or other internal company security documents that define specific policies and standards.

For generating the capability concept, the model of the semiotic triangle will be used. The semiotic triangle is a model in semiotics and linguistics that distinguishes between the sign (or symbol), the thought (or reference), and the referent (or object) while also defining the relationships between them [Ja13]. The creation of the capability concept is an activity where the thought is generated with their attributes to be filled out in further steps of the method, i.e., the thought embodies the specification or description of the symbol "drilling" with their attributes that's needs to be filled out, e.g., name, properties, constraints, parent relationship and composition.

After the capability concept is created and the capability symbol is defined as well as added to the global capability name list, a capability description at different hierarchical levels can be required. One reason for this may be that required capabilities are described in a more abstract form, while offered capabilities are described more specifically. Consequently, different levels of modeling depths are necessary for matching within capability-based engineering. In order to describe the capabilities at different hierarchical levels, it is necessary to identify the capability generalization. Therefore, the upper class (superclass) must be linked to the capability to be described by using the aforementioned activity of identifying existing capabilities. In this activity of identifying generalized capabilities, it is possible that multiple generalizations may be identified either symbolically, which need to be described formally, or else identified with a complete capability description. With a successful identification of generalized capabilities, they have to be referenced. In the case that neither a capability is identified symbolically nor a complete capability description, the aforementioned activity to identify norms, standards, sources, or references is used to introduce a new entry into the global capability name list and to connect an upper-class capability from these other sources to the capability to be described.

A capability can also be composed of other capabilities. The composition of capabilities is a concept that involves combining two or more capabilities to create a new capability. Analog to the activity of identifying the capability generalization, the aforementioned activity of identifying existing capabilities and the activity to identify norms, standards, sources, or references can be used to identify capability compositions.

Properties are used to describe a capability in more detail. These Properties cover a broad spectrum from physical dimensions to sustainability data. To derive a suitable set of properties for a given capability, documents and manuals of production resources can be referred to as well as norms and standards such as IEC 61360 or ECLASS.

Capabilities can be constrained by either property constraints or transition constraints. Both constraint types can be specified by a condition. These conditions are either pre-, post- or invariant conditions. The property constraint is defined by a property value statement, which specifies the context in which a capability may be applied. For example, a capability “Brick burning” might only be applicable in a temperature range of 950°C – 1000°C. Additionally, the oven temperature is to reach before and held while the bricks are burned. Therefore, the property constraint applies pre (before) and invariant (while) conditions on the property temperature. The transition constraint allows to describe a dependency of a capability on another capability. For example, to cut a thread, it is first necessary to drill a hole. Therefore, when utilizing the capability “Threading”, a precondition for the capability “Drilling” might be apply.

Finally, semantic formalization which enables matching of required capabilities (derived from, e.g., product specification or process descriptions) and offered capabilities (assigned to production resources) has to be provided. Therefore, the formalized capability description is used for the following methods to either assign a capability to a production resource or else to derive a capability form a product specification or a process description.

3.2 Assigning Capabilities

Assigning capabilities to operational resources is a systematic approach for manufacturers or operators of a machine, to derive resource capability model instances, in order to allow for advanced automated production planning. In Fig. 3, the workflow of the method Assigning Capabilities with their activities and artifacts is shown.

The first activity involves identifying production resources. In the scope of this activity, the description of the production resource and their environment (if necessary) is used for selection. For acquiring the description, machine-readable descriptions, e.g., formal information models such as an AAS, or resource non-formal documentation, e.g., operation manuals, are used. The information can be gathered by using internal and external sources, e.g., manufacturers product catalogue.

Typically, resources already possess skills, which allows to operate them for the processing of products. Hence, the identification of resource skills becomes paramount, as skills serve as the executable implementations of encapsulated functions specified by capabilities [Di22]. As a prerequisite to the following activities, resource skill descriptions are made available by using the skill descriptions, e.g., by reading information models, such as Module Type Package, PackML or Control Components. The information gathered from the skill description is used to extract further information to determine the effect on the product by processing it in the production resource.

It is essential to ascertain the effect of each resource so that the alternative solutions of the production machines can be abstracted to a capability via a description (of the effect). Therefore, one goal of the production resource is to impose an effect onto the product through skill execution. The effect is described as a change between two states[11], e.g., the change of a product geometry or a temperature change in a process.

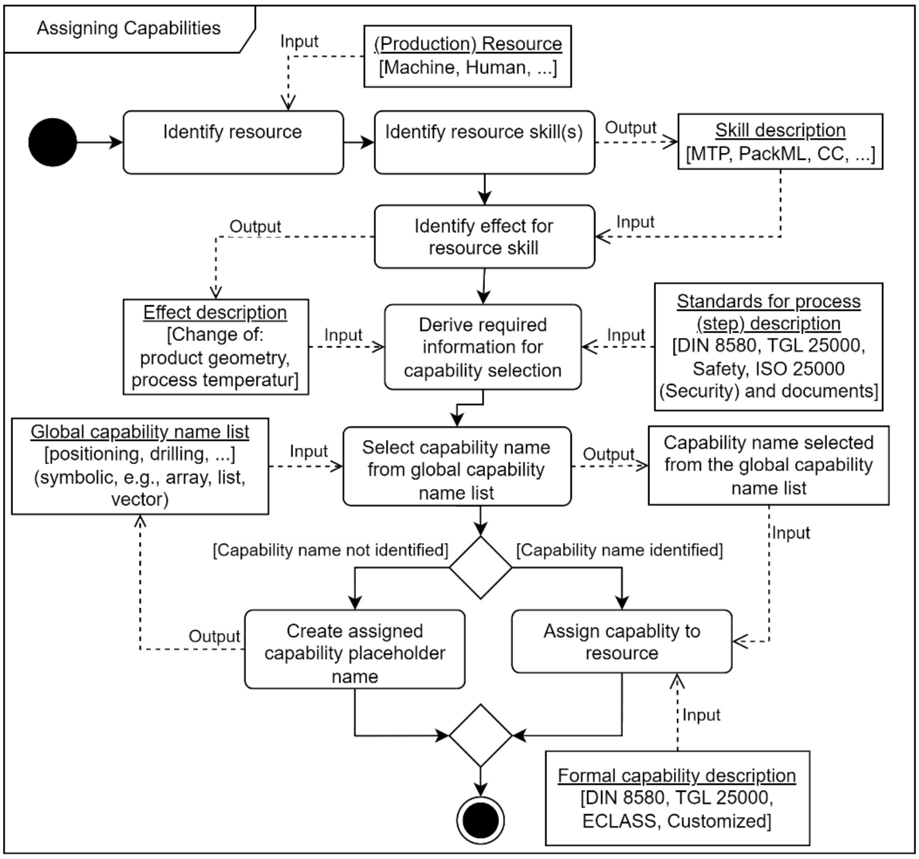

Fig. 3: Workflow for the method of Assigning Capabilities

For the selection of the capability to be assigned, further information to derive it is required in addition to the effect description. This information is gathered from standards or other documents describing processes or operations such as DIN 8580 or TGL 25000. Prepared with the derived information, the capability is to be selected from the global capability name list, as explained before. In addition, the capability name to select, is to ascertain whether the capability aligns with the resource's behavior and effect, as well as whether the capability is described sufficiently for the resource. Either the capability name is identified for an effect, or the capability name is not sufficient.

[11] The effect, as a formal element, is currently under research.

If the capability name for an effect is not identified, a new capability placeholder name has to be created in the global capability name list and it has to be assigned to the production resource. The placeholder is used as a provisional entity until a formal capability description for the given resource is provided by the method Describing Capabilities, see Section 3.1. In case that the capability name for an effect is identified and is also formally described, the capability can be directly assigned to the production resource using the capability name form the global capability name list. For the use of the formal capability description, the selected capabilities for the production resource are made accessible in a capability description model, e.g., the AAS capability description submodel or a resource ontology.

This method of assigning capabilities ensures a structured alignment with resources, fostering an effective utilization of own or external contractor assets, enabled by optimized allocation of these resources.

3.3 Deriving Capabilities

In addition to the two underlying methods described above, the third method focuses on extracting required capabilities from existing descriptions of production procedures or product properties, i.e. product specifications or process descriptions consisting of a logical sequence of process steps for manufacturing the product. The workflow of the method Deriving Capabilities is shown in Fig. 4.

A product specification or process description is necessary to derive required capabilities, which may be available in various forms, such as a computer-aided design (CAD) model, a digital twin, a product data model, a recipe, a procedure, a formal description of a process or other technical specifications. However, these specifications or descriptions alone are not always sufficient to fully identify the required capabilities and their sequence of execution for manufacturing a product.

After identifying the specifications and descriptions, a decision is made whether capabilities have to be derived from a product specification or from a process description. The product can have a modular structure. Consequently, the presence of product parts must be identified. A product part itself is a component of a product to be manufactured. Therefore, a product part is considered equivalent to a product in this contribution.

The objective of the identified products is to ascertain the effects on the product, thereby enabling the derivation of information about possible capabilities for manufacturing the product via a description of the effect. An effect imposed on a product can be, to give a specific example, the change imposed by drilling a desired hole into a piece of metal. The effect description is an artefact that will be used for deriving required information with regard to the capability selection. The next activity in the workflow identifies the process(es) either directly from process description or by the process(es) identified from product specification to be manufactured. Analogously to the product parts, a process can be structured into process steps, which also have to be identified. Existing recipes, procedures or formal descriptions of the process allow the identification of individual process steps. Afterwards, the effects of the process(es) (process steps) have to be identified. For example, an effect that has to be imposed onto the process and the product can be their temperature which has to be within a certain range in order to drill the desired hole in the piece of metal.

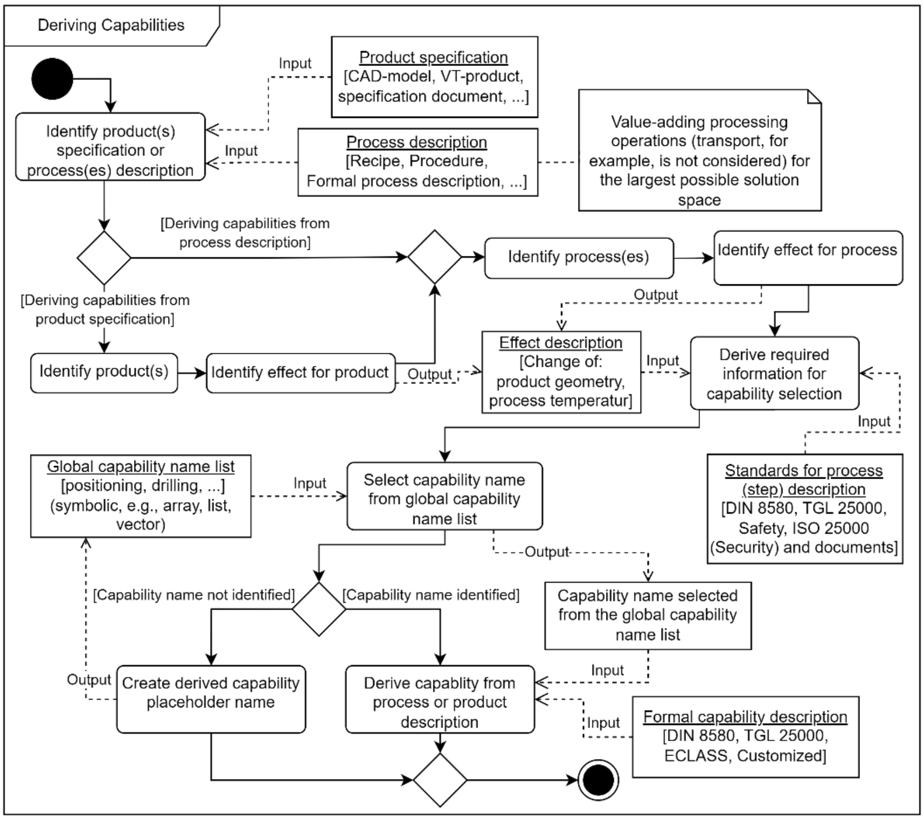

Fig. 4: Workflow for the method of Deriving Capabilities

The gathered information, including that pertaining to the identified effects as described in the effect description and to process-relevant information from standards such as DIN 8580 or TGL 25000, must be considered in order to derive the required information for capability selection. Analogously to the method Assigning Capabilities, the capability has to be selected from the global capability name list. Additionally, the capability name must be selected to determine whether the capability aligns with the effect of the product specification or the process description. The selection of a capability from the global capability name list is dependent on whether the capability id is identified as sufficient or insufficient for an effect. In correspondence to the Assigning Capabilities method, if the capability name for an effect is not identified, a new capability placeholder name is to create in the global capability name list and be the derived capability. Accordingly, the placeholder is to be used as a provisional entity until a formal capability description for the derived capability is provided using the Describing Capabilities method, see Section 3.1. In cases where the name of a capability that imposes the desired effect is identified and also formally described, the capability can be directly derived from the product specification or process description. The method of deriving capabilities ensures a structured derivation of capabilities from product specifications or process descriptions.

4 Conclusion and Outlook

Facing rapidly evolving environments in manufacturing requires flexibility in production. The use of the CSS model as a means to implement Capability-based engineering enables decoupling of product design and production planning and execution. This contribution provides methods for the formal description of capabilities, their assignment to resources, and their derivation from product specifications or process descriptions. The methodical approach through the Capability Determining Method proposes a structure to encompass the description, assignment und derivation of capabilities. This technology-independent method guidelines stakeholders for navigating the complexities of capability-based engineering.

Our research identifies topics, which has to be validated by the working group of Plattform Industrie 4.0. A pilot implementation for the interaction with the CSS model as well as the validation of the methods is currently under development. Therefore, tool support is needed, also to involve various stakeholders. In addition to the current capability concept, the concept of effects was identified and included into the methods to some extent. The concept of effects will be elaborated in further publications.

References

[Di22] Diedrich, C. et.al.: Information Model for Capabilities, Skills & Services, Plattform Industrie 4.0, Berlin, 2022, https://www.plattform-i40.de/IP/Redaktion/DE/Downloads/Publikation/CapabilitiesSkillsServices.html, accessed: 20.01.2024.

[Fr22] Froschauer, R.; Köcher, A.; Meixner, K.; Schmitt, S.; Spitzer, F.: Capabilities and Skills in Manufacturing: A Survey Over the Last Decade of ETFA, 2022 IEEE 27th International Conference on Emerging Technologies and Factory Automation (ETFA), Stuttgart, Germany, pp. 1-8, 2022.

[Gr23] Grüner, S. et. al.: Towards asset administration shell-based continuous engineering in process industries, at-Automatisierungstechnik; 71(8), S. 689-708, De Gruyter (O), Berlin, 2023.

[HDM21] Huang, Y.; Dhouib, S. and Malenfant,J.: An AAS Modeling Tool for Capability-Based Engineering of Flexible Production Lines. In IECON 2021 – 47th Annual Conference of the IEEE Industrial Electronics Society, Toronto, Canada, pp. 1-6, 2021.

[Jä19] Järvenpää, E.; Sitala, N.; Hyllli, O.; Lanz, M.: The development of an ontology for describing the capabilities of manufacturing resources. Journal of Intelligent Manufacturing, vol. 30, no. 2, pp. 959–978, 2019.

[Kö20] Köcher, A. et.al.: Automating the Development of Machine Skills and their Semantic Descriptions. In 2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), pp. 1013–11018, 2020.

[Kö23] Köcher, A. et.al.: A reference model for common understanding of capabilities and skills in manufacturing, at–Automatisierungstechnik; 71(2), S. 94–104, De Gruyter (O), Berlin, 2023.

[Su09] Sure-Vetter, Y. et.al.: Ontology engineering methodology. In: Staab, S, Studer, R (Eds.), Handbook on Ontologies, pp. 135–152, 2009.

[Ja13] Jakus, G. et.al.: Concepts, Ontologies, and Knowledge Representation. SpringerBriefs in Computer Science, DOI: 10.1007/978-1-4614-7822-5_1, 2013.