

# Erweiterung eines Ontologie-basierten Produktkonfigurationsmodells mit generalisierter Microservice Architektur für die Entwicklung in einem Produktionsdatenraum

Erik Konietzko

## Abstract

*In diesem Paper wird die Weiterentwicklung eines auf einer Ontologie basierten Demonstrators für die Konfiguration des Produktmodells einer Brennstoffzelle diskutiert. Der Demonstrator ist Teil eines aufzubauenden resilienten Produktionsdatenraums, der Abhängigkeiten zwischen Produktentwicklungs- und Produktionsdomänen abbildet und auf Änderungen in diesen Domänen funktional reagiert. Das Produktmodell des Produktkonfigurators wurde als Ontologie vorbereitet und eine Informations- und Servicearchitektur um ihn herum entwickelt, um das konfigurierte Produkt als Datenraum-Asset vorzubereiten. Der Fokus der Untersuchung liegt auf der Erweiterung der Ontologien des Produktmodells durch Integration von Servicebeschreibungen in einer Meta-Ontologie. Die entwickelten Ontologie-Modelle dienen zur Validierung der Konsistenz instanzierter Assets und der generalisierten Definition von Services zur Interaktion mit der Konfiguration. Der vorgestellte Anwendungsfall dient der Untersuchung, ob und wie eine Meta-Modellierung geeignet ist, Beschreibungen von Assets in einen generischen Kontext zu bringen, um Datenkonsistenz sowie funktionale Abhängigkeiten zu prüfen und sicherzustellen. Es wird gezeigt, wie Ontologie-Modelle für die Integration von Assets verwendet werden können, um sie mit auf sie angewendeten Services zu verknüpfen. Die Untersuchung führte zur Identifikation benötigter Basisdienste für eine solche Integration und zu weiteren Schritten für die Aufbereitung des Demonstrators als Produktzwilling für die Verwendung in auf semantischen Technologien fußenden Datenräumen.*

*Keywords: Ontologie, Systemmodell, SHACL Validierung, Microservice Architektur*

*DOI: 10.25368/2024.EEE.006*

## 1 Motivation und Problemlage

Die Digitale Transformation in der Produktion und Produktenwicklung bringt zahlreiche Herausforderungen mit sich, insbesondere in Bezug auf die Integration von heterogenen Datenbeständen und Systemen (Abdallah, Shehab & Al-Ashaab, 2021; Gogineni et al., 2021). In modernen Entwicklungs- und Produktionsumgebungen werden Daten aus verschiedenen Quellen generiert, darunter Sensoren, Maschinen, Anlagen und beteiligte IT-Systeme. Diese Daten liegen in unterschiedlichen Formaten, Strukturen und Standards vor, was die Integration und Zusammenführung zu einem komplexen Prozess macht. Die Herausforderungen bestehen darin, diese heterogenen Datenbestände zu verstehen, zu harmonisieren und in einer einheitlichen Weise zu

integrieren. Ebenso stellt das Vorhandensein unterschiedlicher Systeme, Datenformate und Technologien eine zentrale Herausforderung dar, Interoperabilität zwischen heterogenen Daten- und Technologiebeständen zu etablieren und zu gewährleisten (Gogineni et al., 2021; Khan & Turowski, 2016; Mountantonakis & Tzitzikas, 2020). Diese Interoperabilität ist jedoch entscheidend, um die Effizienz und Flexibilität von Prozessen zu verbessern und innovative Geschäftsmodelle mit heterogenen Daten zu ermöglichen. (Curry, Scerri & Tuikka, 2022) beschreiben Datenökosysteme und Datenräume als einen vielversprechenden Ansatz in der aktuellen Forschung, um heterogene Daten über unterschiedliche Domänen und Unternehmungen hinweg zu integrieren und die Möglichkeit zur Entwicklung intelligenter Services zu schaffen. Sie erklären weiterhin die Notwendigkeit von semantischen Technologien, um heterogene Daten und Services in einem vereinheitlichten Datenraumkontext interpretierbar zu machen.

Der Entwicklungsschwerpunkt dieser Untersuchung liegt auf der Analyse von semantischen Beschreibungen unterschiedlicher Systeme und Datenmodelle und ihre Erweiterbarkeit für die Integration funktionaler Services. Um die erweiterte Beschreibung durch Ontologien und eine Prüfung dieser für die Eignung zur Verknüpfung von Datenraum Assets zu untersuchen, wurde in einer vorhergehenden Entwicklung eine Brennstoffzelle für einen Anwendungsfall „Produktkonfiguration“ semantisch beschrieben (vgl. (Konietzko & Gogineni, 2024)). In dieser Untersuchung wurde die Anwendung durch semantische Servicebeschreibungen innerhalb einer Microservice-Architektur für Service-Assets erweitert. Einerseits dienen diese für die Interaktion der Benutzer in der Konfiguration, um abhängige Parameter innerhalb des Produktmodells basierend auf der definierten Semantik zu berechnen. Andererseits liefern die Serviceerweiterungen des Produktmodells domänenübergreifende und konfigurationsabhängige Informationen aus Fertigungs- und Montageprozessen, wie z. B. Fräskonturen oder Logistikinformationen.

## **2 Stand von Wissenschaft und Technik**

In Datenrauminitiativen, wie Catena-X, Gaia-X, International Data Space oder der Asset Administration Shell (AAS) werden Datenräume als digitale Umgebungen entwickelt, die darauf abzielen, semantische Interoperabilität zu ermöglichen und den effizienten Austausch von Daten und Diensten zwischen verschiedenen Parteien zu erleichtern. Sie dienen als Plattform für die Zusammenarbeit und den Informationsaustausch zwischen Akteuren aus verschiedenen Domänen und Unternehmen. Datenräume ermöglichen die Integration heterogener Daten und fördern eine effiziente und transparente Zusammenarbeit entlang der Wertschöpfungskette, wodurch die Grundlage für einen souveränen und vertrauenswürdigen Datenhandel zwischen

verschiedenen Domänen geschaffen wird (Hagelien, Preisig, Friis, Klein & Konchakova, 2021; Solmaz et al., 2022). Zu den Herausforderungen in der Ausgestaltung von Datenräumen gehören die Heterogenität der Daten (unterschiedliche Domänen und Datenquellen), die semantische Interoperabilität (klare Definition von Bedeutung und Beziehungen) und die Granularität der Beschreibung (hier wirkt der Detailgrad für eine verbesserte Interpretierbarkeit gegen die Flexibilität generischer Beschreibungen) (Gelhaar & Otto, 2020; Obrst et al., 2014; Solmaz et al., 2022).

(Meta-)Ontologien spielen eine wichtige Rolle bei der Vereinfachung der semantischen Beschreibung von Datenraum Assets. Sie dienen als abstrakte Modelle, um die Beschreibung fachspezifischer Domänenmodelle zu erweitern, zu generalisieren und zu standardisieren (Ernadote, 2015). Für die Integration heterogener Datenbestände finden Ontologien im industriellen Kontext vielfache Ansätze und Verwendung; vergleichend können hierzu, unter anderen, (Ekaputra, Sabou, Serral Asensio, Kiesling & Biffli, 2017) oder (Noy, 2004) genannt werden. Durch die Verwendung von Meta-Ontologien können Datenraum Assets unabhängig von ihrer spezifischen Domäne oder Struktur in einer einheitlichen Weise beschrieben werden. (Gelhaar & Otto, 2020; Obrst et al., 2014; Solmaz et al., 2022) beschreiben wie diese die Integration und den Austausch von Assets erleichtern und eine effiziente Nutzung ermöglichen. Weiter erklären sie, dass (Meta-)Ontologien dazu beitragen, die Herausforderungen bei der semantischen Beschreibung von Datenraum Assets zu bewältigen und die Interoperabilität und Flexibilität in Datenräumen zu verbessern.

### **3 Fragestellung und Forschungsziel**

In diesem Paper wird die Verwendung von (Meta-)Ontologien untersucht, um die Interaktion von Services und Datenmodellen durch semantische Beschreibungen zu ermöglichen. Durch die Beschreibung digitaler Assets, wie Daten-, Service- oder Kommunikationsmodelle, in einer gemeinsamen Ontologie werden diese heterogenen Ökosystemen interpretierbar. In (Konietzko & Gogineni, 2024) wird der grundlegende Aufbau eines Demonstrators zur Konfiguration eines ontologischen Brennstoffzellen-Produktmodells dargestellt. In diesem Paper wird die Erweiterung der Ontologie-Kontexte des Produktmodells mit Serviceinformationen und die Eignung von Ontologien zur Abbildung und Umsetzung funktionaler Abhängigkeiten untersucht. Es konzentriert sich tiefergehend auf die Erweiterung von Integrationsmechanismen zwischen Service-Assets und dem Produktmodell, unter Berücksichtigung von Potenzialen, Voraussetzungen und Herausforderungen bei der Entwicklung einer funktionalen Servicearchitektur auf Grundlage der semantischen Produktbeschreibung. Dabei wird die Konfiguration der Brennstoffzelle als Produktmodell-Beispiel weiter ausgebaut und ein generisches Meta-Ontologie-Modell verwendet, um das Produktmodell mit Servicebeschreibungen zu erweitern. Der

Generalisierungsansatz untersucht, wie Meta-Ontologien die funktionale Integration von Daten und Service-Assets erleichtern können, mit dem Ziel, den Beschreibungsaufwand zu reduzieren und vereinheitlichte Konzepte für die Interoperabilität zu identifizieren.

## 4 Lösungsansatz und Methodik

### 4.1 MBSE Orientierte Entwicklung

In diesem Kontext wurde trotz der Ontologie-zentrierten Ausrichtung der Architektur ein MBSE-Ansatz verfolgt, um eine umfassende Erfassung von Abhängigkeiten und die Berücksichtigung von Modellierungsausprägungen in Bezug auf Parameter und Servicearchitektur sicherzustellen. Dieser Ansatz dient dazu, die Entwicklung des Configurators methodisch abzusichern. Die Vorgehensweise in der Entwicklung ist an das MBSE V-Modell nach (Sünnecioglu, Brandenburg, Caspar, Fauquembergue & Dörr) angelehnt, um die Entwicklung des Configurators sowohl als eigenständigen Teil als auch als integralen Bestandteil des Produktionsdatenraums methodisch abzusichern. Die vorgenommenen Schritte sind in Abbildung 7 (links) skizziert.

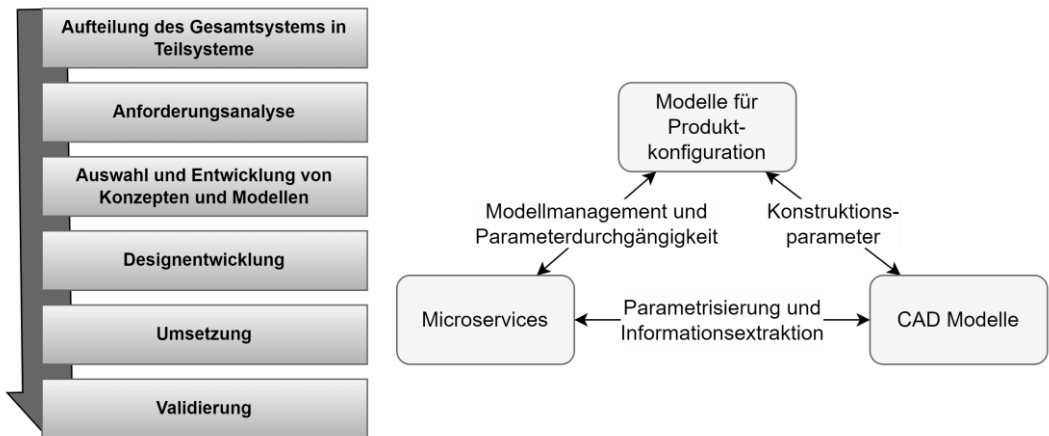


Abbildung 7: Methodisches Vorgehen (links) und sich ergebende Abhängigkeiten zwischen zu implementierenden Modellen und Services (rechts)

Die Anforderungsanalyse des Systems führte zur Identifikation von grundlegenden Modellen und Konzepten, welche für das zu entwickelnde Design implementiert wurden (vgl. Abbildung 7 (rechts)). Zum einen ist ein entsprechendes Modell notwendig, um die Informationen der Konfiguration darzustellen und gebündelt zu integrieren (Modelle für Produktkonfiguration). Zum anderen sollen entsprechende Konfigurationsparameter automatisiert in den CAD Modellen der Zusammenbaugruppe abgebildet werden und sich ergebende Änderungen im CAD Modell an die Modelle der Produktkonfiguration zurückgegeben werden. Um

Konsistenz und Durchgängigkeit von Informationen sicherzustellen werden Funktionen (Microservices) benötigt, um mit den unterschiedlichen Modellen zu interagieren und den benötigten Informationsfluss zwischen den Modellen zu koordinieren und zu gewährleisten.

Als Implementierungsansatz wurde eine Microservice-Architektur gewählt, um die Entwicklung zu erleichtern und externe Abhängigkeiten zu koordinieren. Die Aufteilung in Subsysteme aus dem methodischen Vorgehen ermöglichte eine einfache Integration von externen Abhängigkeiten zu Domänen im Produktionsdatenraum. Die Microservice-Architektur ist ebenso auf die Integration in einen Datenraum ausgerichtet, in dem unterschiedliche Daten und Service Assets über eine vereinheitlichte Semantik verständlich und interoperabel werden. Darüber hinaus erleichtert die Verwendung von Microservices die Erweiterung und Weiterentwicklung der Anwendungsfallservices und Architektur.

## 4.2 Ontologischer Backbone

Das Produktmodell wurde als Ontologie für die Konfiguration modelliert und mit einer Meta-Ontologie für generische System- und Serviceausprägungen kombiniert. Um die Konsistenz instanzierter Konfigurationen mit den vorgegebenen Restriktionen der Meta-Ontologie zu überprüfen, wurden Validierungsmechanismen implementiert. Dabei wurde die Shapes Constraint Language (SHACL), eine ontologische Validierungssprache, verwendet. Mit Hilfe von sogenannten SHACL Shapes können die erwarteten Datenstrukturen mittels Constraints definiert werden. Informationen in einer Ontologie können so mithilfe der SHACL Shapes auf korrekte Modellierung validiert werden. (Knublauch & Kontokostas, 2017) Beispielsweise kann in einer Validierung geprüft werden, ob die im Produktmodell Layer als „MustParameter“ (eine Meta-Ontologie Klasse, vgl. Abbildung 8) klassifizierte Parameter tatsächlich instanziiert werden, wenn dies durch eine entsprechende SHACL Shape vorausgesetzt wird. Dieser Ansatz wurde in Anlehnung an die Ausgestaltung von Datenräumen gewählt. In diesen werden Assets unterschiedlicher Provenienz so beschrieben und überprüft, dass sie in konkreten Anwendungskontexten der Datenräume, wie etwa Mobilität oder Produktion, gesichert interpretierbar sind. Der methodische Aufbau der Ontologie erfolgte nach dem Vorgehen in (Konietzko & Gogineni, 2023). Konietzko und Gogineni gehen genauer auf die Ausgestaltung der Meta-Ontologie und die Prüfmechanismen für den Aufbau valider Konfigurationen ein (Konietzko & Gogineni, 2024). Dieser Ansatz wird weiterverfolgt und auf funktionale Abhängigkeiten mittels zu prüfender (REST-) Servicebeschreibungen in der Meta-Ontologie erweitert, um die generalisierten Asset-Beschreibungen mit Services für die Produktkonfiguration zu verknüpfen.

Abbildung 8 zeigt die grundlegende Struktur für die Validierung über Meta-Ontologien. Der Meta Layer enthält die Meta-Ontologie mit generalisierten Konzepten in „System-Subsystem“, „Parameter“ und „Service“ Kategorien mit unterschiedlichen Subklassifikationen. Diese generischen Meta-Konzepte können in der Verwendungsebene im Produktmodell Layer auf explizite Anwendungsfälle konkretisiert werden. Innerhalb des Produktmodell Layer liegt die kontextualisierte Beschreibung einer Brennstoffzellen-Ontologie für die Konfiguration vor. In diesem Layer werden explizite Daten und Services aus dem Daten und Service Layer ontologisch beschrieben und instanziiert. Der Produktmodell Layer integriert damit Service und Daten Assets für die Produktkonfiguration unter Verwendung der Konzepte des Meta Layer, wodurch Prüfservices und Validierungsmechanismen basierend auf SHACL anwendungsfallsspezifisch angewendet werden können.

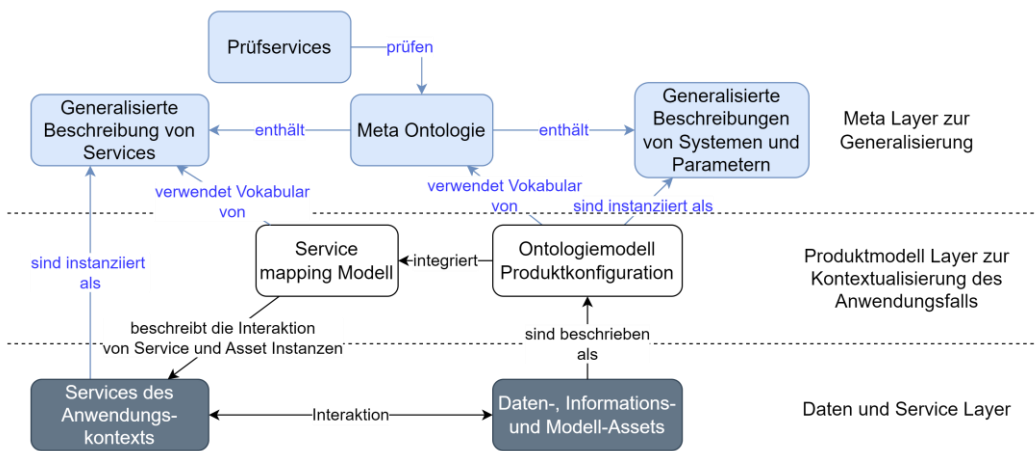


Abbildung 8: Ontologisches Layer Modell

### 4.3 Ontologische Modellierung von Microservices

Die Metamodellierung wird an der generalisierten Beschreibung und Einbindung von Microservices im Meta Layer veranschaulicht (vgl. Abbildung 9). Die Meta-Klasse „Service“ ist in unterschiedliche Service Kategorien subklassifiziert. Im Weiteren wird sich auf die Subklasse „ServiceFunction“ fokussiert. Diese beschreibt Services, welche als externe Webservices integriert werden sollen. Eine „ServiceFunction“ kann grundsätzlich näher annotiert werden, etwa mit einem Namensattribut oder Kommentaren. Eine „ServiceFunction“ hat Subjekt-, Endpoint-, Input- und Output-Properties. Diese können wiederum als optional oder verpflichtend klassifiziert werden, um eine korrekte Funktion von Serviceinstanzen durch die SHACL Validierung zu gewährleisten. Als Subjekt wird eine Klasse definiert, an der ein Service ausgeführt werden soll. Die Inputs beschreiben dabei jene Attribute in der Ontologie, die als Request Parameter an den Service Endpoint übergeben werden. Über die Definition

der Outputs kann die Antwort des externen Service koordiniert und entsprechende Informationen als Property in der Ontologie instanziiert werden. In einem „hasMapping“ Property der „ServiceFunction“ Klasse kann optional eine Eineindeutigkeit von Inputs und Outputs von Services spezifiziert werden, falls als „Input“ und „Output“ definierte Properties für mehrere Klassen in einer Ontologie bereitstehen (z. B. das generische Attribut „rdfs:label“). Ein entsprechender Interpretationsservice wurde für die Mapping Ausdrücke implementiert.

Ein Modellierungsbeispiel eines Webservices zur Informationsextraktion von Stücklistendokumenten (BOM.csv) einer Konfiguration ist in Abbildung 9 dargestellt.

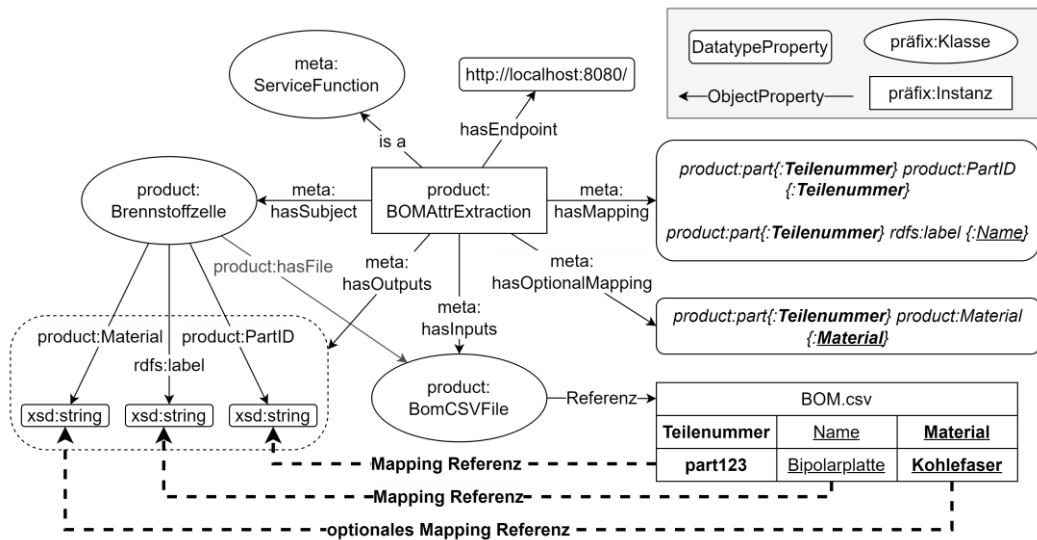


Abbildung 9: Mapping Beispiel eines Service zur Informationsextraktion aus einer Stückliste

Die Vokabeln der Ontologien werden mit ihren jeweiligen Präfixen gekennzeichnet. Die Präfixe „rdfs:“ und „xsd:“ referenzieren Standard-Ontologien. Das Präfix „meta:“ verweist auf Konzepte aus der Ontologie des Meta Layer und das Präfix „product:“ verweist auf das Modell im Produktmodell Layer. In der Abbildung ist die Klasse „Brennstoffzelle“ mit den Properties „Material“, „PartID“ und „label“ skizziert, welche mit einer Klasse „BomCSVFile“ verbunden ist. Diese referenziert auf das explizite Stücklistendokument der Brennstoffzelle. Auf die Darstellung, durch welche Meta-Konzepte die Klassen „BomCSVFile“ und „Brennstoffzelle“ sowie ihre Properties beschrieben sind, wurde der Übersichtlichkeit wegen verzichtet. Für den Service der Informationsextraktion ist das Individual „BOMAttrExtraction“ als Instanz der Meta-Klasse „ServiceFunction“ angelegt und verwendet die Properties dieser. Als Input des Service ist die „BomCSVFile“ Klasse mit Referenz zum Stücklistendokument definiert.

Die Outputs sind die dargestellten Properties der „Brennstoffzelle“ Klasse. Die „BOMAttrExtraction“ hat im gegebenen Beispiel zwei Mapping Properties, wovon eines optional ist. Diese Properties beinhalten Mapping Ausdrücke, mit deren Hilfe es möglich ist Informationen aus der Stückliste, die der Service bereitstellt, als die jeweiligen Properties der Klasse „Brennstoffzelle“ zu interpretieren. Das „hasMapping“ sagt aus, dass die Information „Teilenummer“ als eine Instanz „part{Teilenummer}“ (z. B. „part123“) der Klasse „Brennstoffzelle“ angelegt wird und ebenso das Property „PartID“ diesen Wert enthält. Die Information „Name“ (z. B. „Bipolarplatte“) aus der Stückliste wird im Beispiel für das Property „rdfs:label“ verwendet. Das „hasOptionalMapping“ Property sagt aus, dass die „Material“ Information (z. B. „Kohlefaser“) aus der Stückliste, sofern vorhanden, als „Material“ Property der jeweiligen Brennstoffzelle angelegt wird. Sollte eine Information nicht bereitstehen, wird ein optionales Mapping ignoriert.

## 5 Ergebnisse

Basierend auf der dargestellten Layer Struktur wurde ein Produktkonfigurator für Brennstoffzellen-Produktmodelle erweitert. Hierbei ermöglichen die Modelle des Meta Layer eine generische Beschreibung von Services, ohne eine konkrete Funktionsbeschreibung zu verwenden. Dadurch können Services generisch angesprochen werden (z. B. als Webapplikation) und im Produktmodell Layer näher spezifiziert werden, wie z. B. ein Service zur Parametrisierung von CAD Modellen verschiedener Konfigurationen. Somit wird eine automatisierte Koordination externer Services über die Ontologie-Modelle ermöglicht.

Die grundlegende System- und Service Architektur des Produktkonfigurators ist in Abbildung 10 dargestellt. Die Modelle für die Konfiguration sind in den Datenspeichern hinterlegt. Ein CAD Grundmodell steht im Objektspeicher zur Parametrisierung bereit. Die Ontologie-Modelle sind in einer Graph Datenbank gespeichert, welche Repositories für die Meta-Ontologie, SHACL Shapes zur Konsistenzprüfung, das Produktmodell der Brennstoffzelle, die Servicebeschreibungen inklusive Mappings und Konfigurationsinstanzen enthält. Die Interaktionsservices stellen grundlegende Funktionalitäten, wie CRUD-Operationen, für die Modellinteraktion in den Speichern bereit. Über ein User Interface kann die Konfiguration der Brennstoffzelle vorgenommen werden (1). Das Interface verwendet das Produktmodell der Brennstoffzelle, um die Konfigurationsparameter abzufragen oder neue Services zu integrieren. Die Konfiguration wird zur Validierung an den Service SHACL Validierung übergeben (2). Korrekte Konfigurationen werden im entsprechenden Repository als Instanzen gespeichert (3). Basierend auf den Servicebeschreibungen werden die integrierten Webservices koordiniert und ausgeführt (4). Für die Konfiguration werden Services zur Parametrisierung und Erstellung des CAD Modells (5) sowie der Service



zur Extraktion von Stücklisteninformationen angesprochen. Die erstellten Modelle werden im Objektspeicher abgelegt (6) und die definierten Outputs zur Konfigurationsinstanz hinzugefügt (7).

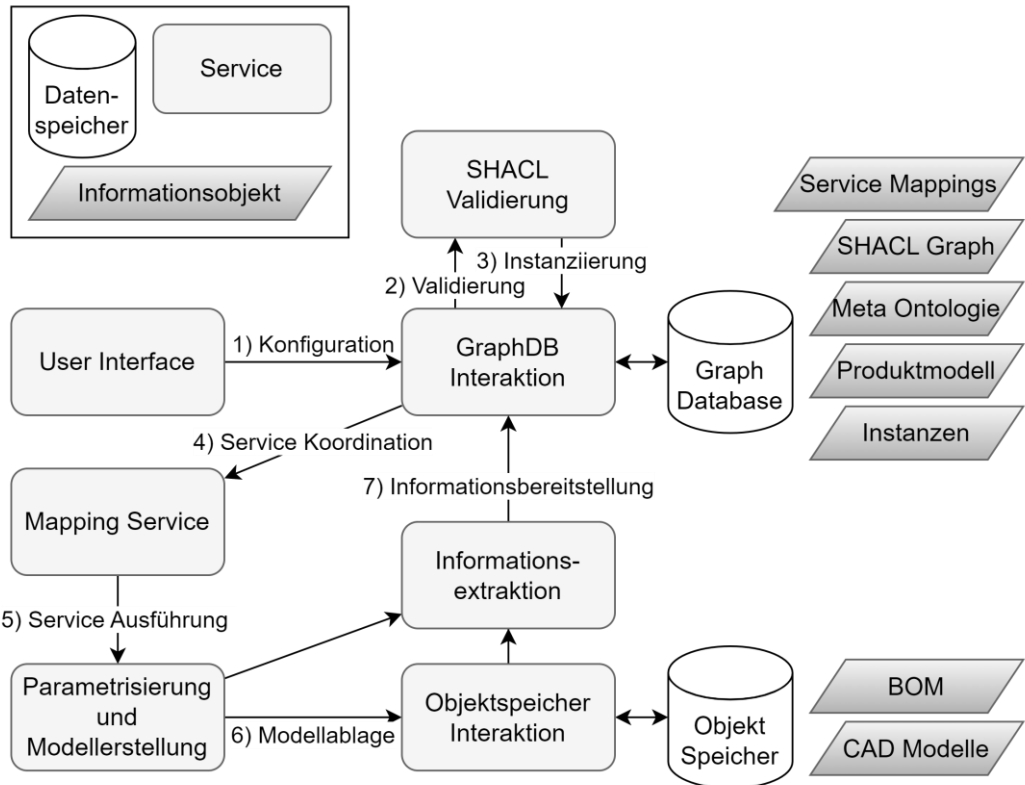


Abbildung 10: System- und Service-Architektur des Demonstrators

Die entwickelte Service-Architektur kann für den Informationsfluss innerhalb der Konfiguration in Basisdienste und Anwendungsdienste unterteilt werden. Anwendungsdienste charakterisieren einen spezifischen Anwendungsfall und sind speziell auf diesen zugeschnitten, während Basisdienste eine grundlegende Funktionalität bieten, um bspw. Informationsflüsse zu koordinieren. Für die Produktkonfiguration sind die Anwendungsdienste die Services zur Parametrisierung und Modellerstellung, sowie der Service zur Extraktion von Stücklisteninformationen. Die Basisdienste sind die Interaktionsservices mit den Datenspeichern und der Mapping Service zur Koordination weiterer Webservices und Informationsinterpretation durch Mappings. Der SHACL Validierung Service ist für den Informationsfluss nicht zwingend erforderlich, aber er ist ein integraler Bestandteil des Meta-Ontologie Ansatzes und gewährleistet die Konsistenz von Produkt- und

Servicebeschreibungen. Daher wird er als Basisdienst eingestuft. Da die Basisdienste für die grundlegende Funktionalität des Produktkonfigurators verantwortlich sind, sind sie nicht in den Ontologie-Modellen beschrieben.

## 6 Diskussion und Ausblick

Die Untersuchung zeigt, dass Meta-Ontologien für die Generalisierung von Konzepten wie Daten und Services geeignet und verwendbar sind. Die Vorteile ontologischer Modelle, wie schnelle Modellierung und einfache Integration, sprechen für diesen Ansatz. SHACL ermöglicht es, Restriktionen über die Meta-Ontologie in den Produktmodell Layer einzubringen, um die korrekte Funktionalität sicherzustellen. Es wurden jedoch auch Herausforderungen identifiziert, wie ein notwendiges tiefergehendes Verständnis der Meta-Ontologie für die Modellierung des Produktkonfigurationsmodells und der manuelle Aufwand bei der Modellierung von Domänenmodellen.

Zudem ist die Implementierung von Basisservices zur Koordination funktionaler Abhängigkeiten notwendig. Im Anwendungsfall wurden entsprechende Services identifiziert, welche für eine Grundfunktionalität zwischen ontologischen Modellen und Serviceangeboten vorgehalten werden müssen. Diese umfassen die Interaktion mit verwendeten Datenspeichern, Konsistenzprüfung von beschriebenen Assets (SHACL Validierung), Services zur Interaktion mit Ontologie-Modellen sowie Transformationsmechanismen (Mapping oder Übersetzung) für unterschiedlich geartete Datenmodelle. Die Entwicklung solcher Basisdienste erfordert zunächst Mehraufwand, erleichtert jedoch auf Nutzungsebene die Beschreibung der Assets für funktionale Interoperabilität. Dieser Mehraufwand, z. B. durch Definition von Mappings, fällt jedoch bei jeder Änderung und Erweiterung der Modelle aus dem Meta- oder Produktmodell Layer an. Die Erkenntnisse aus dem Anwendungsfall führen zu weiteren Untersuchungen zur Integration und zum Aufbau in einem Datenraum. Die Erweiterung des Produktmodell Layer auf andere Domänen ist erforderlich, um den Ansatz auch domänenübergreifend zu validieren und die entsprechenden Entwicklungsaufwände bewerten zu können. Ebenso bedeutet dies weitere Untersuchungen zur Abstraktion von Assets für die Genese von Meta-Vokabeln. Generell ist die Erweiterung der Ontologie-Modelle und Mappings mit etablierten (Ontologie-) Modellen, Standards und Technologien zu untersuchen und dahingehende Automatisierungsmechanismen zu prüfen.

Eine Herausforderung besteht in der Einbindung zusätzlicher heterogener Daten-, Service- und Beschreibungsmodelle. Für die Entwicklung eines ganzheitlichen Produktionsdatenraums ergeben sich weitere Schritte, wie die Analyse von Vor- und Nachteilen zwischen Ontologie-Modellen und SysMLv2-Modellen in der Umsetzung funktionaler Änderungen zwischen Modellen. Zukunftsweisend werden die Modelle in

einem Produktionsdatenraum integriert. Hierfür müssen Abhängigkeiten zu Beschreibungen und Funktionen des Datenraums identifiziert und die Modellierung der Ontologie Modelle dahingehend angepasst werden. Hier ist eine Analyse zur Verwendung und Integration vorhandener standardisierter Assetformate, wie Submodule der Verwaltungsschale (AAS), erforderlich.

## Literaturverzeichnis

Abdallah, Y. O., Shehab, E. & Al-Ashaab, A. (2021). Digital transformation challenges in the manufacturing industry. In M. Shafik & K. Case (Eds.), *Advances in manufacturing technology XXXIV* (pp. 9–14). Derby: IOS Press.

Curry, E., Scerri, S. & Tuikka, T. (2022). *Data Spaces*. Cham: Springer International Publishing.

Ekaputra, F., Sabou, M., Serral Asensio, E., Kiesling, E. & Biffi, S. (2017). Ontology-based data integration in multi-disciplinary engineering environments: a review. *Open Journal of Information Systems*, 4(1), 1–26.

Ernadote, D. (2015). An ontology mindset for system engineering. In *2015 IEEE International Symposium on Systems Engineering (ISSE)* (pp. 454–460). Piscataway: IEEE.

Gelhaar, J. & Otto, B. (2020). Challenges in the emergence of data ecosystems. In *Proceedings of the 24th Pacific Asia Conference on Information Systems, PACIS 2020*. New York: AIS Electronic Library.

Gogineni, S., Brünnhäuber, J., Lindow, K., Konietzko, E. P., Stark, R., Nickel, J. et al. (2021). Systematic design and implementation of a semantic assistance system for aero-engine design and manufacturing. *International Journal of Metadata, Semantics and Ontologies*, 15, 87–103. <https://doi.org/10.1504/IJMSO.2021.120281>

Hagelien, T., Preisig, H., Friis, J., Klein, P. & Konchakova, N. (2021). A practical approach to ontology-based data modelling for semantic interoperability. In *Proceedings of the 14th WCCM-ECCOMAS Congress (vol. 10)*. CIMNE.

Khan, A. & Turowski, K. (2016). A survey of current challenges in manufacturing industry and preparation for Industry 4.0. In A. Abraham, S. Kovalev, V. Tarassov & V. Snášel (Eds.), *Proceedings of the First Interational Scientific Conference "Intelligent Information Technologies for Industry" (IITI'16)* (pp. 15–26). Basel: Springer.

Knublauch, H. & Kontokostas, D. (2017). *Shapes Constraint Language (SHACL)*. <https://www.w3.org/TR/shacl/>, abgerufen am 31.01.2024.

Konietzko, E. P. & Gogineni, S. (2023). Ontology based skill matchmaking between contributors and projects in open source hardware. In E. Garoufallou & A. Vlachidis (Eds.), *Metadata and Semantic Research. 16th Research Conference, MTSR 2022, London, UK, November 7–11, 2022, Revised Selected Papers* (pp. 14–25). Cham: Springer Nature Switzerland.

Konietzko, E. P. & Gogineni, S. (2024). Ontologie-basiertes Konfigurationsmanagement für Systemmodelle. In D. Wilke, W. Koch, R. Kaffenberger & S. Dreiseitel (Hrsg.), *Tag des*

Systems Engineering 2023. Tagungsband Würzburg, 15.-17. November 2023 (S. 227–233). Bremen: Gesellschaft für Systems Engineering.

Mountantonakis, M. & Tzitzikas, Y. (2020). Large-scale semantic integration of linked data. *ACM Computing Surveys*, 52(5), 1–40. <https://doi.org/10.1145/3345551>

Noy, N. F. (2004). Semantic integration. *ACM SIGMOD Record*, 33(4), 65–70. <https://doi.org/10.1145/1041410.1041421>

Obrst, L., Gruninger, M., Baclawski, K., Bennett, M., Brickley, D., Berg-Cross, G. et al. (2014). Semantic web and big data meets applied ontology. *Applied Ontology*, 9(2), 155–170. <https://doi.org/10.3233/AO-140135>

Solmaz, G., Cirillo, F., Fürst, J., Jacobs, T., Bauer, M., Kovacs, E. et al. (2022). Enabling data spaces. In N. Laoutaris & M. Mellia (Hrsg.), *Proceedings of the 1st International Workshop on Data Economy* (pp. 42–48). New York: Association for Computing Machinery.

Sünnetcioglu, A., Brandenburg, E., Caspar, J., Fauquembergue, M. & Dörr, S.. Information Retrieval Techniken zum Automatisierten Erstellen von Tracelinks im Model Based Systems Engineering. *Digitales Engineering zum planen, testen und betreiben technischer Systeme*, 48–53.

## **Kontakt**

Erik Paul Konietzko, M. Sc.

Fraunhofer Institut für Produktionsanlagen und Konstruktionstechnik IPK

Pascalstraße 8-9

10587 Berlin

[www.ipk.fraunhofer.de](http://www.ipk.fraunhofer.de)