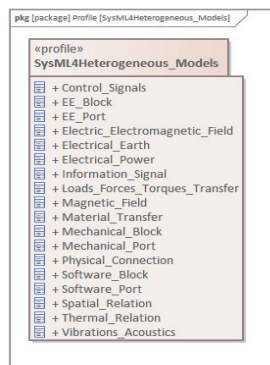


## Erzeugung von SysML-Stereotypen zur Beschreibung logischer Systemarchitekturen im Model-based Systems Engineering

Schumacher, T.; Inkermann, D.

*Aktuelle Systementwicklungen erfordern disziplinspezifische und disziplinübergreifende Modellierungstechniken. Für disziplinübergreifende Modellierungstechniken wird häufig die Systems Modeling Language (SysML) genutzt. Da es sich bei der SysML um eine universell einsetzbare Modellierungssprache handelt, ist diese für konkrete Entwicklungssituationen oft sehr abstrakt. Dieser Beitrag erläutert die gezielte Erweiterung der SysML durch Stereotypen und wendet diese für die Erstellung heterogener Produktmodelle an. Außerdem werden zukünftige Forschungsarbeiten vorgestellt.*



*Current system developments require both discipline-specific and cross-domain modelling techniques. Cross-domain modelling techniques preferably use the Systems Modeling Language (SysML). Since SysML is a universally applicable modelling language, it is often very abstract for specific development situations. This article explains the targeted extension of SysML using SysML element stereotypes and their application within heterogeneous product models. Additionally, future research fields will be presented.*

### Einleitung und Problemstellung

Moderne Systeme sind durch ein dynamisches Verhalten, starke Vernetzung mit Nachbarsystemen, Anbindung an das Internet und die Nutzung von On-Demand-Diensten gekennzeichnet /1/. Die Entwicklung dieser Systeme erfordert die Kollaboration unterschiedlicher Entwicklungsdisziplinen, welche üblicherweise auf disziplinspezifische Entwicklungsmodelle zurückgreifen. Eine erfolgreiche Systementwicklung bedarf zusätzlich einer gesamthaften Systembetrachtung, die durch disziplinübergreifende Modellierungstechniken unterstützt wird /2/. Dazu existieren eine Vielzahl verschiedener Modellierungstechniken, siehe /3/, welche zumeist auf die Modellierungssprache *Systems Modeling Language* (SysML) zurückgreifen. Bei der SysML handelt es sich um eine grafische und universell einsetzbare Modellierungssprache, die für jegliche Systeme, wie Automobil- oder Luft- und Raumfahrtssysteme, Anwendung finden kann /4/. Die universellen Einsatzmöglichkeiten der SysML stellen gleichzeitig auch Herausforderungen für die Anwendung in einer konkreten Domäne dar, da die allgemeingültigen Modellierungssprachen oft die konkreten Fragestellungen nicht im notwendigen

Detaillierungsgrad abbilden können. Um dieser Problemstellung zu begegnen, bietet die SysML die Möglichkeit spezifische Elementstereotypen und Profile für konkrete Anwendungsfälle zu definieren.

### Zielstellung und Aufbau dieses Beitrags

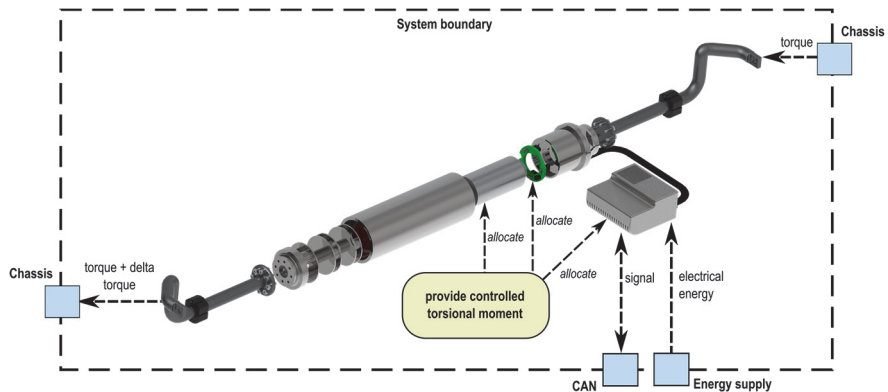
Dieser Beitrag betrachtet die Erweiterung der SysML durch Definition von Elementstereotypen zur Erzeugung heterogener Produktmodelle. Die Anwendung heterogener Produktmodelle und die Integration von Elementstereotypen unterstützt bspw. bei der Beschreibung logischer Systemarchitekturen mechatronischer Systeme, aufgrund einer detaillierteren Beschreibung der Systemelemente und Schnittstellen. Dazu werden im folgenden Abschnitt der Begriff des heterogenen Modells und die Vorgehensweisen zur Erzeugung von SysML-Stereotypen erläutert. Anschließend werden allgemeingültige SysML-Modellelemente identifiziert und erforderliche Erweiterungen für den Anwendungsfall *Beschreibung logischer Systemarchitekturen mechatronischer Systeme* erarbeitet. Abschließend werden die erzeugten Modellelementstereotypen an einem Beispiel in Form eines heterogenen Modells angewendet.

### Grundlagen und Stand der Technik

Dieser Abschnitt erläutert die Anwendung heterogener Produktmodelle innerhalb des Model-based Systems Engineerings (MBSE) und stellt die notwendigen Grundlagen der Erzeugung von SysML-Stereotypen vor.

### Anwendung heterogener Produktmodelle

Heterogene Modelle bieten die Möglichkeit Partialmodelle oder Modellelemente unterschiedlicher Modelltypen in eine Modellpräsentation zu integrieren /5/. Beispielsweise können Funktionen, Wirkprinzipien, Systemelemente sowie Relationen und Anforderungen in einem Modell zusammengeführt werden /6/. Abbildung 1 stellt ein beispielhaftes heterogenes Modell vor, welches sowohl Modellelemente der SysML als auch CAD-Partialmodelle beinhaltet.



**Abbildung 1:** Visualisierung eines heterogenen Modells, aus /2/

Zielstellung heterogener Produktmodelle ist es, unterschiedliche Modellelemente für konkrete Fragestellung zu kombinieren, um aussagekräftige Modelle zu generieren. Das heterogene Produktmodell, dargestellt in Abbildung 1, adressiert zwei konkrete Fragestellungen.

1. Die Integration einer Systemgrenze, sowie notwendige Schnittstellen und Relationen zwischen SysML-Elementen und CAD-(Partial)Modellen ermöglichen eine detaillierte Beschreibung der Systemschnittstellen. Dies kann beispielsweise während der Integration des Systems in das übergeordnete System (in diesem Beispiel: das Fahrzeugchassis) unterstützen.
2. Die Integration funktionaler Produktbeschreibungen (gelbes Oval) und deren Allokation zu den realisierenden Systemelementen unterstützt bei der funktionalen und räumlichen Produktpartitionierung.

Die Anwendung heterogener Modelle für konkrete Fragestellungen innerhalb des Model-based Systems Engineerings erfordert die Erweiterung der allgemeinen SysML-Modellelemente. Daher werden im folgenden Abschnitt eine Vorgehensweise zur Erzeugung von Elementstereotypen eingeführt und anschließend ausgewählte SysML-Elemente für den Anwendungsfall *Beschreibung logischer Systemarchitekturen* erweitert.

### **Vorgehensweise zur gezielten Erweiterung der SysML für konkrete Anwendungsfälle**

Die SysML schlägt für die Modellierung von Systemanforderungen, Systemverhalten und Systemaufbau verschiedene Modell- bzw. Diagrammtypen sowie Modellelemente vor. Für eine gezielte Erweiterung der SysML werden sogenannte Stereotypen verwendet. Stereotypen erweitern bestehende SysML-

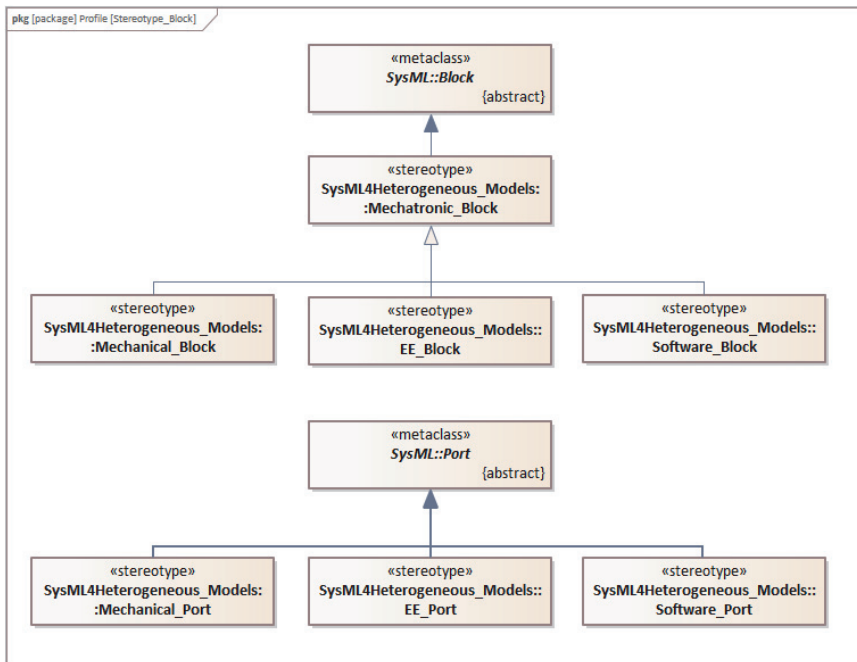
Modellelemente um zusätzliche Eigenschaften (Properties) sowie (Zwangs)Bedingungen (Constraints) und werden in Profilen gruppiert /4/. In einem Profil werden die konkreten Elementstereotypen mit den klassischen SysML-Modellelementen, welche eine jeweilige Metaklasse darstellen, verknüpft. Das konkrete Vorgehen zur Erweiterung der SysML durch Stereotypen kann bspw. bei Friedenthal et al. /4/ oder Weilkiens /7/ eingesehen werden.

### Erzeugung von SysML-Elementstereotypen zur Beschreibung logischer Systemarchitekturen

Dieser Beitrag wendet das Vorgehen zur Erzeugung von SysML-Stereotypen für den Anwendungsfall *Beschreibung logischer Systemarchitekturen* an. Eine logische Architektur beinhaltet die dekomponierten logischen Systemelemente und beschreibt die grundsätzliche Systemstruktur /8/. Zentrale Grundlage für eine logische Architektur sind die Systemfunktionen (funktionale Architektur), welche das Systemverhalten beschreiben. Innerhalb der Architekturentwicklung erfolgt eine Zuweisung der Systemfunktionen zu den logischen Systemelementen, welches eine wichtige Grundlage für spätere Realisierungsentscheidungen darstellt. Demnach sollten logische Architekturen möglichst lösungsneutral beschrieben sein.

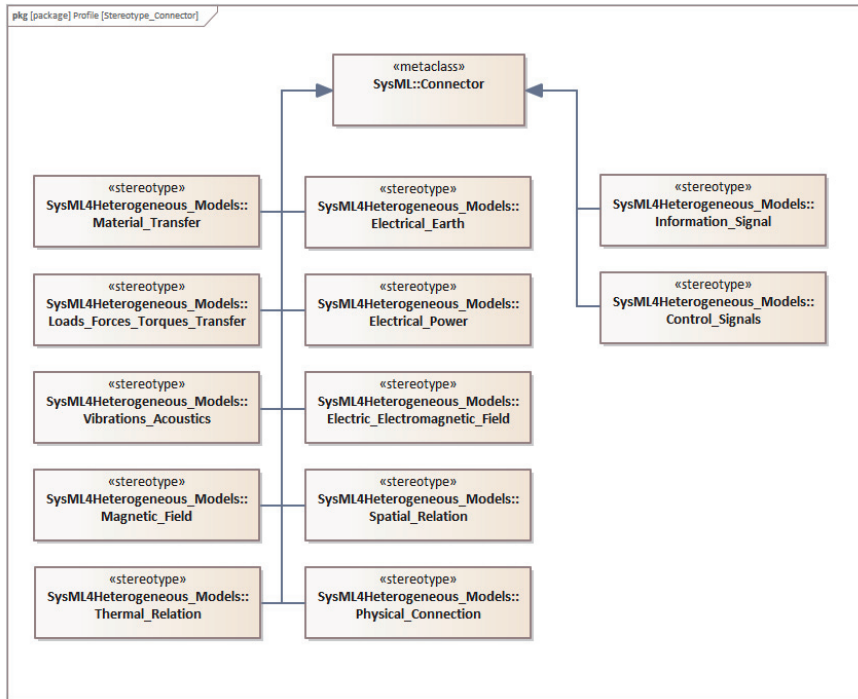
Zur Beschreibung einer logischen Systemarchitektur können folgende SysML-Elemente verwendet werden: SysML::Block, SysML::Port und SysML::Connector. Ein SysML::Block beschreibt dabei ein Systemelement einschließlich dessen Eigenschaften. SysML::Ports dienen der Beschreibung von Schnittstellen zwischen den Systemelementen und zur Umwelt. Eine Beschreibung der Interaktionen bzw. Relationen zwischen Systemelementen kann beispielsweise durch das Element SysML::Connector erfolgen.

Für eine Beschreibung der Systemstruktur einschließlich der Schnittstellen ist es für mechatronische Systeme wesentlich eine disziplinspezifische Zuordnung der Systemelemente (SysML::Block) und der Schnittstellen (SysML::Port) vorzunehmen, da je nach Disziplin unterschiedliche Elementeneigenschaften beschrieben werden müssen. Daher wurden für das Modellelemente SysML::Block Stereotypen für die Mechatronik-, Mechanik-, Elektrik/Elektronik- und Softwaredisziplin angelegt. Gleichzeitig erfordert eine erfolgreiche Entwicklung mechatronischer Systeme auch eine detaillierte Beschreibung der System(elemente)schnittstellen. Um eine detaillierte Schnittstellenbeschreibung einschließlich der erforderlichen Schnittstelleneigenschaften vornehmen zu können, wurde das Element SysML::Port ebenfalls um disziplinspezifische Stereotypen (Mechanik, Elektrik/Elektronik und Software) erweitert. Die eingeführten Stereotypen sind in der Abbildung 2 visualisiert.



**Abbildung 2:** Erzeugung von disziplinspezifischen SysML::Block und SysML::Port Stereotypen

Zur Beschreibung einer logischen Systemarchitektur, müssen zusätzlich die Relationen zwischen den Systemelementen berücksichtigt werden. Hierfür stellt die SysML beispielsweise den Verknüpfungstyp SysML::Connector bereit. Um die Relationen zwischen disziplinspezifischen Systemelementen (SysML::Block) und Schnittstellen (SysML::Port) im ausreichenden Detaillierungsgrad beschreiben zu können, werden, basierend auf /9/, verschiedene Verknüpfungsarten unterschieden und als SysML::Connector Stereotypen angelegt, siehe Abbildung 3.



**Abbildung 3:** Erweiterung SysML::Connector um ausgewählte Stereotypen

Anhand der eingeführten Modellelementstereotypen kann eine detaillierte Beschreibung der Systemarchitektur vorgenommen werden. Der Folgeabschnitt stellt dies anhand eines Anwendungsbeispiels dar.

### Anwendung von SysML-Elementstereotypen zur Beschreibung der logischen Systemarchitektur am Beispiel elektromechanische Wankstabilisierung

Die Evolution einer logischen Systemarchitektur basiert auf den definierten Systemanforderungen und einer funktionalen Systembeschreibung /10/. Abbildung 4 stellt daher diese verschiedenen Modellelemente beispielhaft in einer Präsentation (Modell) am Beispiel der elektromechanischen Wankstabilisierung dar. Gleichzeitig beinhaltet das Modell die eingeführten Modellelementstereotypen, um die Schnittstellen zwischen dem System und der Umwelt zu beschreiben.

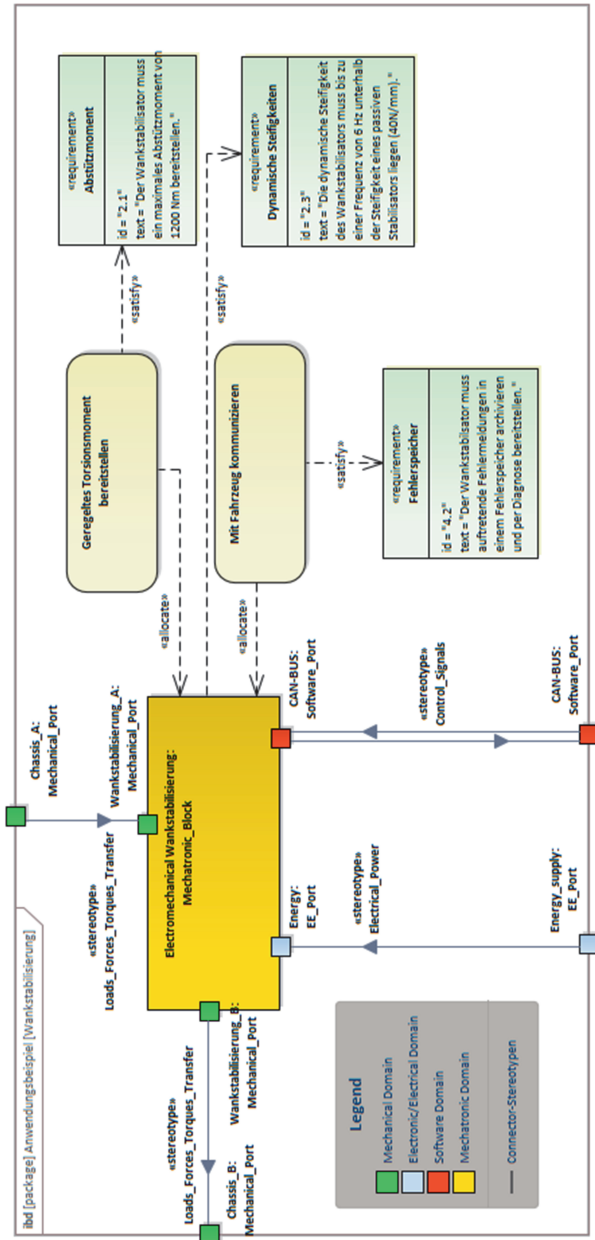


Abbildung 4: Heterogenes Produktmodell am Beispiel Wankstabilisierung

Entsprechend der Erläuterung (Abschnitt *Anwendung heterogene Produktmodelle*) stellt Abbildung 4 ein heterogenes Systemmodell dar, da unterschiedliche Elementtypen (Anforderungen, funktionale- und strukturelle Systembeschreibungen) in eine Präsentation integriert sind. Diese abstrakte Beschreibung bildet eine gute Grundlage, um die logischen Systemelemente und deren Abhängigkeiten zu entwickeln. Abbildung 5 visualisiert die logische Systemarchitektur am Beispiel elektromechanische Wankstabilisierung und greift dabei die erarbeiteten Elementstereotypen auf.

Die Einführung der Stereotypen unterstützt bei der Definition und Abgrenzung von Systemelementen sowie bei der konkreten Beschreibung der Relationen. Außerdem hilft diese Modellierungsform bei der Identifikation weiter zu dekomponierender Systemelemente (z. B. Systemelement *Steuergerät* in Abbildung 5) und bei der Festlegung von Verantwortlichkeiten im Rahmen des Entwicklungsprozesses. Die zentrale Zielstellung für die Einführung der Modellstereotypen ist jedoch eine detaillierte Beschreibung der einzelnen Modellelemente, bspw. durch disziplinspezifische Eigenschaften oder Schnittstellenbeschreibungen, um ausgewählte Entwicklungsaktivitäten innerhalb des Model-based Systems Engineerings zu unterstützen.

## Zusammenfassung

Dieser Beitrag erläutert die Notwendigkeit der gezielten Erweiterung der *Systems Modeling Language* (SysML) innerhalb des Model-based Systems Engineerings. Hierzu wird eine Vorgehensweise zur Erweiterung der SysML durch Erzeugung von Modellelementstereotypen und Anwendung heterogener Produktmodelle erläutert. Anhand des Anwendungsfalls *Beschreibung logischer Systemarchitekturen mechatronischer Systeme* wurden konkrete SysML-Elementstereotypen definiert und diese anhand eines Anwendungsbeispiels angewendet. Außerdem wurde die Anwendung und der Nutzen heterogener Produktmodelle grundlegend dargestellt. Zukünftige Forschungsarbeiten fokussieren insbesondere die Vermeidung bzw. den Umgang mit Modellinkonsistenzen im MBSE. Dazu werden zunächst grundlegend verschiedene Typen von Modellinkonsistenzen und etablierte Lösungsansätze recherchiert. Anschließend soll die automatisierte Erzeugung heterogener Modelle, anhand technischer Schnittstellen (API) und Modellverknüpfungen das Auftreten von Modellinkonsistenzen im Model-based Systems Engineering reduzieren.



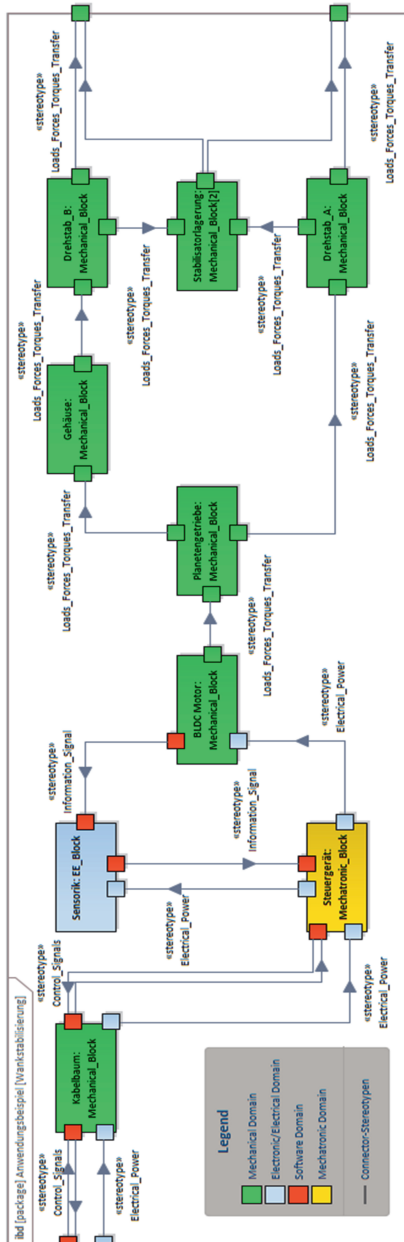


Abbildung 5: Logische Systemarchitektur am Beispiel Wankstabilisierung

## Danksagung

Der vorliegende Beitrag ist Teil der Arbeiten des niedersächsischen Zukunftslabors Mobilität. Das Teilprojekt wird gefördert vom Niedersächsischen Ministerium für Wissenschaft und Kultur (Fördernummer ZN3493) im Niedersächsischen Vorab der VolkswagenStiftung und betreut vom Zentrum für digitale Innovationen Niedersachsen (ZDIN).

## Literatur

- /1/ VDI/VDE 2206: Development of mechatronic and cyber-physical systems, Beuth Verlag, Düsseldorf, 2021
- /2/ Schumacher, T., Inkermann, D.: Heterogeneous models to Support Interdisciplinary Engineering – Mapping Model Elements of SysML and CAD, Proceeding of 32th CIRP Design Conference, Paris, 2022
- /3/ Kernschmidt, K.: Interdisciplinary structural modeling of mechatronic production systems using SysML4Mechatronics, Dissertation, TU München, 2019
- /4/ Friedenthal S. et al.: A Practical Guide to SysML, MK/OMG Press, Amsterdam, 2017
- /5/ Jansen, S., Welp, E. G.: A heterogeneous modelling approach for domain allocation in mechatronics, Proceedings of 16th International Conference on Engineering Design (ICED 2007), Paris, 2007
- /6/ Schumacher T., Inkermann D.: Heterogene Modellierung - Verknüpfung und Integration von Systemmodellen der SysML mit CAD-Modellen, Proceedings of 32th DfX-Symposium, 2021
- /7/ Weilkiens T.: Systems engineering with SysML/UML - Modeling, analysis, design, MK/OMG Press, Amsterdam, 2007
- /8/ Walden D.D., et al.: Systems engineering handbook, Wiley, Hoboken, 2015
- /9/ Wilms R., et al.: Identifying Cross-Domain Linkage Types to Support Engineering Change Management and Requirements Engineering, Proceedings of 29th CIRP Design, Póvoa de Varzim, 2019
- /10/ Kleiner S., Kramer C.: Model Based Design with Systems Engineering Based on RFLP Using V6, Proceedings of Smart Product Engineering, 2013