

Schlussbericht

SET Level

SET Level

Simulationsbasiertes Entwickeln und Testen von automatisiertem Fahren

Beitrag des Zuwendungsempfängers: **PROSTEP AG**
Dolivostraße 11
64293 Darmstadt

Laufzeit: 01.03.2019 – 31.10.2022

Das diesem Bericht zugrunde liegende Vorhaben wurde mit Mitteln des Bundesministeriums für Wirtschaft und Klimaschutz unter dem Förderkennzeichen 19A19004H gefördert.

Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Version: 1.0
Stand: 18.04.2023



Bundesministerium
für Wirtschaft
und Klimaschutz

Inhaltsverzeichnis

0	Kurze Darstellung	4
0.1	Aufgabenstellung	4
0.2	Voraussetzungen für das Vorhaben	5
0.3	Planung und Ablauf	6
0.4	Stand der Wissenschaft und Technik	8
0.5	Zusammenarbeit mit anderen Stellen	9
1	Eingehende Darstellung	10
1.1	Erzielte Ergebnisse	10
1.1.1	Teilprojekt 1: Use-Case Management und funktionsorientierte Anforderungsanalyse	10
1.1.2	Teilprojekt 2: Simulationsbasiertes und virtuelles Entwickeln / Testen.....	14
1.1.3	Teilprojekt 3: Modellspezifikation, -Entwicklung und –Validierung	23
1.1.4	Teilprojekt 4: Instanziierung von Werkzeugketten für Entwicklung und Testen ...	35
1.1.5	Teilprojekt 5: Einbettung und kritische Reflexion	88
1.2	Zahlenmäßiger Nachweis	107
1.3	Notwendigkeit und Angemessenheit der geleisteten Arbeit	107
1.4	Nutzen und Verwertbarkeit	107
1.5	Bekannt gewordener Fortschritt	108
1.6	Veröffentlichungen	109
1.6.1	Erfolgte Veröffentlichungen	109
2	Abbildungsverzeichnis	111
3	Tabellenverzeichnis	114
4	Literaturverzeichnis	115

Geschlechtergerechte Sprache

Die in diesem Dokument verwendeten Personenbezeichnungen beziehen sich immer gleichermaßen auf Personen jeglichen Geschlechts. Auf eine Doppelnennung und gegenderte Bezeichnungen wird zugunsten einer besseren Lesbarkeit verzichtet.

0 Kurze Darstellung

0.1 Aufgabenstellung

PROSTEP unterstützte die Projektpartner bei der Anforderungserhebung für die Simulation sowie für die Szenarienräume und Referenzszenarien für den identifizierten Anwendungsfall der „komplexen urbanen Kreuzung“. Zusätzlich beteiligte sich PROSTEP an der Sammlung und übersichtlichen Darstellung von Standards, die im Projektkontext von Relevanz sind. Ein weiterer Arbeitsschwerpunkt war die Erstellung eines glaubwürdigen (credible) Simulationsprozesses. In Zuge dessen wurde ein Prozessframework definiert und erstellt. Es umfasst unter anderem die beiden selbstständigen Prozesse, Credible Simulation und Credible Modelling Process, sowie den Vereinbarungsprozess und das Simulation bzw. Modelling Request. Spezieller Wert wurde dabei auch auf die Nachvollziehbarkeit und Traceability mit Fokus auf eine partnerübergreifende Zusammenarbeit gelegt. Zusätzlich dazu wurde ein Kooperationsprozess, zwischen Entwicklungspartnern, in Anlehnung an den SmartSE-Referenzprozess [1] definiert, der wiederum die beiden „credible“ Prozesse unterstützt. Des Weiteren beteiligte sich PROSTEP an der Sammlung der Anforderungen an die exemplarische Integrationsarchitektur. Darüber wirkte PROSTEP an der Entwicklung und dem Aufbau einer exemplarischen integrativen Simulationsarchitektur mit. Für die einheitliche Darstellung der entwickelten integrativen Simulationsarchitektur erstellte PROSTEP ein UML-Modell der Architektur nach dem RFLP-Ansatz über die verschiedenen Architekturebenen in Enterprise Architect. Weiterhin wurde sichergestellt, dass die beschriebene Architektur zum im Projekt entwickelten Credible Simulation Process kompatibel ist. Ebenso wirkte PROSTEP an der Erstellung und Beschreibung der Metadatenbeschreibung für Modelle basierend auf den zuvor identifizierten Anforderungen mit. Ein weiterer Arbeitsschwerpunkt von PROSTEP war die Entwicklung und der Einsatz eines Traceability-Tools. Dieses wurde während der Projektlaufzeit den Partnern zur Nutzung zur Verfügung gestellt und entsprechend den Anforderungen aus dem Projekt kontinuierlich erweitert und optimiert. Zu den wichtigsten Funktionalitäten zählen dabei u.a. das Anlegen und Suchen von Artefakten (inkl. Metadaten) sowie der Trace-Links zwischen Artefakten und deren grafische Darstellung, das Abbilden des Credible Simulation Processes durch Templates, die Möglichkeiten zur Durchführung dynamischer Impact-Analysen (Filter), die Realisierung einer ersten Baselining-Funktionalität und die Cloudfähigkeit der TRACY-Software. PROSTEP wirkte ebenfalls an der Erstellung einer prozesssicheren Gesamtbeschreibung mit, welche für die Dokumentation der Simulation Use Cases auf Basis des im Projekt entwickelten Credible Simulation Process verwendet werden kann. Die zugehörige Dokumentation wurde im von PROSTEP entwickelten Traceability Tool durchgeführt. Eine übergreifende Aufgabe von PROSTEP war die Unterstützung der Projektpartner bei der Durchführung des „Proof of Concept“ zur projektinternen Validierung der wesentlichen Projektergebnisse. PROSTEP erstellte einen generischen Einführungsprozess für die industrielle Instanziierung der Projektergebnisse bei den beteiligten Industriepartnern. Weiterhin unterstützte PROSTEP die industrielle Erprobung der Projektergebnisse. Dazu zählen unter anderem die Erhebung industrieller Anforderungen an hochautomatisierte Fahrzeuge, die industrielle Evaluation des im Projekt entwickelten Kooperationsprozesses und Simulation Request. Die gewonnenen Erkenntnisse wurden in das Projekt zurückgemeldet und zum Teil als „Best Practises“ zusammengeführt. Vor der Rückübertragung der Erkenntnisse der Industriepartner in das Projekt, wurden diese zunächst anonymisiert, um zu vermeiden, dass firmeninterne Erkenntnisse nach außen getragen werden. Darüber hinaus unterstützte PROSTEP den Austausch mit dem Schwesterprojekt VVMethoden sowie den prostep ivip Gremium SmartSE, in dem einige Arbeiten des SET Level Projektes nach Projektende weitergeführt werden.

0.2 Voraussetzungen für das Vorhaben

SET Level wurde als Folgeprojekt von PEGASUS konzipiert. Das Projekt PEGASUS (Projekt zur Etablierung von generell akzeptierten Gütekriterien, Werkzeugen und Methoden sowie Szenarien und Situationen zur Freigabe hochautomatisierter Fahrfunktionen) mit einer Laufzeit von 2016 bis 2019 hatte das Ziel, Basismethoden für die Automatisierungslevel 3 und 4 im Einsatz auf Autobahnen zu entwickeln (siehe Abbildung 1).

- ▶ The **PEGASUS Family** focuses on development / testing methods and tools for AD systems on highways and in urban environments

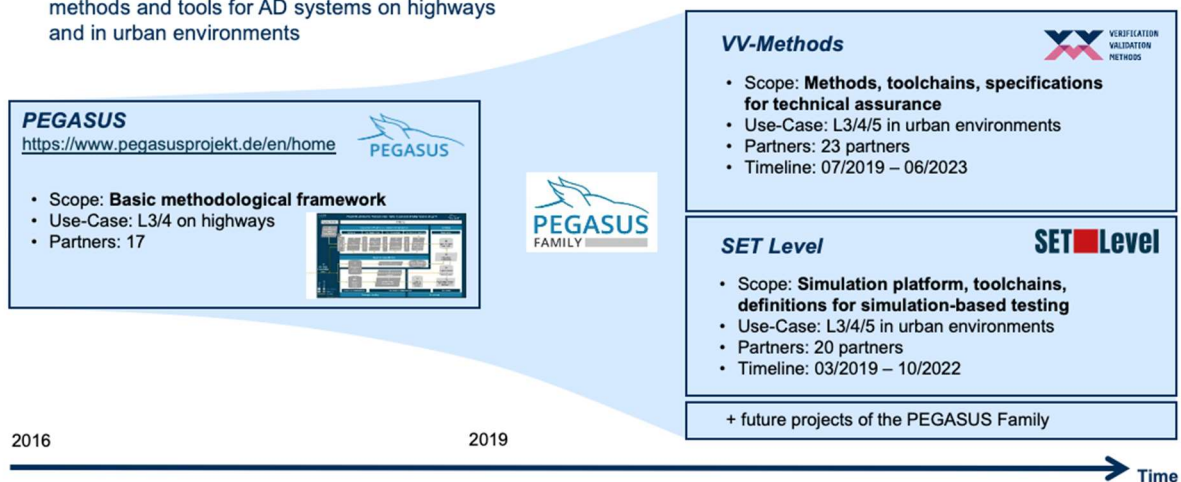


Abbildung 1: Projekte der PEGASUS-Familie

SET Level wurde speziell für den Einsatz von Simulation beim Entwickeln und Testen hoch automatisierter Fahrfunktionen in urbanen Umgebungen konzipiert. Dabei wurden wesentliche Partner aus der deutschen Automobilindustrie, Zulieferer, IT-Vendoren und renommierten Forschungseinrichtungen zusammengeführt, um Kompetenzen, Methoden und Technologien gemeinsam aufzubauen (siehe Abbildung 2).

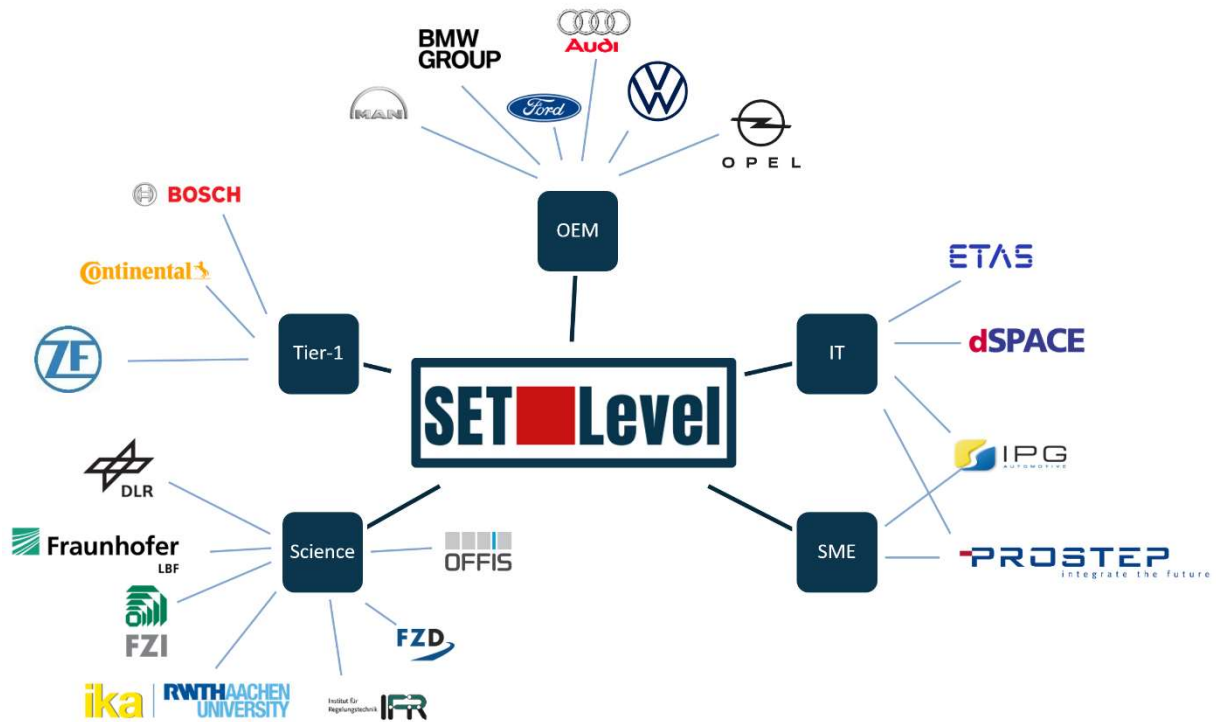


Abbildung 2: Zusammensetzung der Partner von SET Level

0.3 Planung und Ablauf

Grundansatz zur Erschließung und Bearbeitung der komplexen Fragestellung bildete ein iteratives Vorgehen unter schrittweiser Komplexitätssteigerung zur gemeinsamen Durchdringung und Entwicklung des Themas (siehe Abbildung 3).

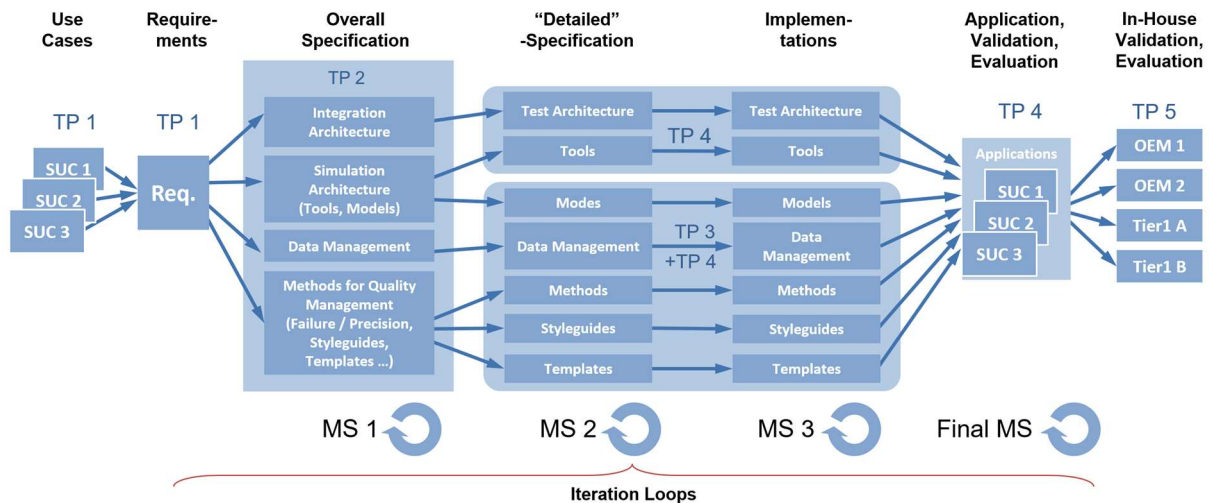


Abbildung 3: Schnelle Iterationen in Projektmeilensteinen

Als Erfolgsmerkmal und -nachweis wurden so genannte Simulation Use Cases (SUC s) konzipiert. Um einen Erfolgsnachweis und eine Basis für die Reflektion der gewählten Ansätze zu schaffen, wurden zu definierten Meilensteinen Demonstratoren implementiert und evaluiert. Dieses Vorgehen war entscheidend für ein schnelles Lernen und für den Nachweis, dass die Grundideen von SET Level, die modularisierte und auf Standards basierenden Architekturen von Simulationstools, auch wirklich funktionieren.

Ausgehend von den Use Cases wurden im Teilprojekt 1 (TP1) Anforderungen (Requirements) an Simulationstools spezifiziert. Inhaltlich fand die Konzeption (Specification) des modularen und standardbasierten Aufbaus von Simulationstools im Teilprojekt 2 (TP2) statt. Die detaillierte Spezifikation notwendiger Prozesse und die Implementierung von Modellen wurde im Teilprojekt 3 (TP3) durchgeführt. Die Ausführung von Simulationen auf unterschiedlichen Simulationstools (von der Implementierung bis zur Evaluation) fand in Teilprojekt 4 (TP4) statt. Die Sicherstellung der industriellen Anwendbarkeit erfolgte in Teilprojekt 5 (TP5) durch regelmäßige Bewertungen des aktuellen Entwicklungsstands sowie entsprechendes Feedback.

BMW und das DLR koordinierten das Projekt als Ganzes. Die Arbeit erfolgte in einer Vielzahl von Teilprojekten (siehe Abbildung 4).

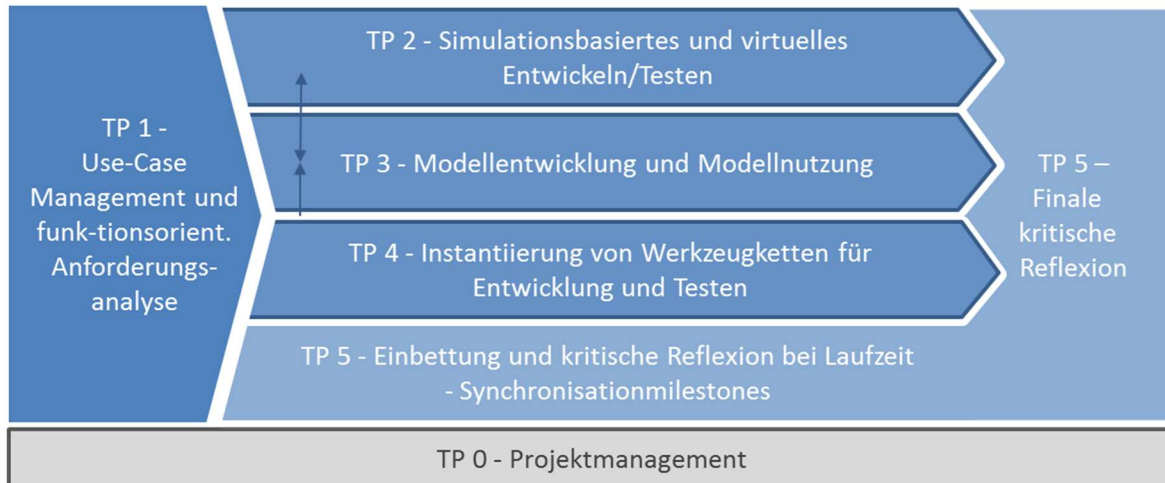


Abbildung 4: Aufbau der Teilprojekte von SET Level

Zeitplan

In

Abbildung 5 ist eine Gesamtübersicht der Projektlaufzeit dargestellt. Die verschiedenen Farben sind qualitative Intensitätsangaben, grob eingeteilt in geringe, mittlere und hohe Kapazitätsbedarfe der jeweiligen Teilprojekte. Wie zu erkennen ist, hatte Teilprojekt 1 als Anforderungs- und Synchronisationsprojekt zu VVMethoden einen starken Fokus im ersten Jahr und benötigte im weiteren Verlauf immer weniger personelle Ressourcen, um die Kommunikation zu VVMethoden zu gewährleisten. Die Teilprojekte 2 und 3 hatten ihre Belastungsspitze beginnend mit Q3 über den größten Teil des zweiten Projektjahres. Teilprojekt 4 hatte einen gleichmäßigen Kapazitätsbedarf über die Projektlaufzeit, während der von Teilprojekt 5 über den Projektverlauf immer höher wurde und gegen Ende am höchsten war.

Abbildung 5 sind auch drei Gesamtmeilensteine jeweils zum Ende eines Projektjahres verzeichnet, deren Hauptergebnisse als Prototyp, Vorversion, Endversion der Werkzeugketten verstanden werden können.

Arbeitspaket	Projektjahr 1				Projektjahr 2				Projektjahr 3				Projektjahr 4		
	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12	Q13	Q14	Q15
0															
1.1															
1.2															
2.1															
2.2															
2.3															
3.1															
3.2															
3.3															
4.1															
4.2															
4.3															
4.4															
5.1															
5.2															
5.3															

Abbildung 5: Zeitplanung mit Meilenstein- und Abschlussberichten

Dabei war als Erfolgsmerkmal die Konzeption von Simulation Use Cases (SUCs) und deren Implementierung in Demonstratoren entscheidend für schnelles Lernen und für den Nachweis, dass die Grundideen von SET Level, die modularisierte und auf Standards basierenden Architekturen von Simulationstools auch wirklich funktionieren.

Als zielführend hat sich die Organisation sog. Quartalsmeetings bewährt, bei denen alle Projektbeteiligten nach vorgegebener Agenda ihre jeweiligen Fortschritte berichtet und das jeweils weitere Vorgehen mit den übrigen Projektpartnern diskutiert und abgestimmt haben.

Im ersten Projektjahr konnten alle Quartalsmeetings durch physische Treffen vor Ort organisiert werden. Erfolgsfaktor war dabei die Einrichtung eines gemeinsamen Repositories für Dokumente, Modelle und weitere Artefakte auf der Basis von GitLab.

Ab dem zweiten Projektjahr waren Corona-bedingt im wesentlichen nur noch virtuelle Meetings möglich. Hier hat die Nutzung von Microsoft Teams und die Einführung von Miro als Kommunikationsplattform erheblich geholfen.

0.4 Stand der Wissenschaft und Technik

Das Projekt SET Level knüpfte an den aktuellen Stand der Wissenschaft und Technik betreffend Methoden und Werkzeuge des Testens automatisierter Fahrzeugfunktionen an. Insbesondere lieferten die Dokumentationen aus den Projekten PEGASUS (Verweis <https://www.pegasusprojekt.de/de/>), ENABLE S3 (Verweis <https://enable-s3.eu/>) und SmartSE (<https://www.prostep.org/en/projects/smart-systems-engineering/>) dafür eine gute Ausgangsbasis.

Die Hauptergebnisse des Projektes PEGASUS sind in der Definition einer Gesamtmethode zur Bewertung einer hochautomatisierten Fahrfunktion zusammengeführt worden. Diese Gesamtmethode umfasst die Aspekte Anforderungsdefinition, Testableitung, Testvorgehen und Werkzeugunterstützung. Bereits dort wird ein besonderer Schwerpunkt auf die Rolle der Simulation gelegt, und die Thematik der Formulierung von Simulationsaufgaben mithilfe der damaligen Versionen der Formate OpenDRIVE und OpenSCENARIO wird eingehend betrachtet. Insbesondere werden dort auch Defizite der Formate für die zweckgemäße Nutzung festgestellt und Vorschläge zur Behebung der Defizite gemacht. Diese Untersuchungen zur Rolle von Simulation in der Absicherung hochautomatisierter Fahrzeugfunktionen und zu den Realisierungsansätzen für eine praktisch nutzbare Anwendung konnten für SET Level aufgegriffen werden und bildeten einen wesentlichen Teil der Basis, auf der das Projekt systematische Lösungen entwickelte.

ENABLE S3 betrachtete Verifikation und Validierung nicht nur im Automobilbereich, sondern auch in Luft- und Raumfahrt, Bahn, Seefahrt sowie Medizin- und Agrartechnik. Das Projekt strebte nach möglichst einheitlichen, übertragbaren Ansätzen, Methoden und Lösungen. So

konnten neben spezifischen automobilbezogenen Ergebnissen (etwa zur Simulationsvalidität) auch generische Beschreibungen in SET Level herangezogen werden, um darauf aufsetzend die automobilspezifischen Ansätze weiter zu entwickeln.

Für SET Level sehr spezifische Konzepte und Ergebnisse konnten aus dem prostep-ivip-Projekt SmartSE herangezogen werden. Die dort entwickelten und erprobten Ansätze zur Austauschbarkeit von Verhaltensmodellen eingebetteter Systeme unter Verwendung des FMI-Standards (Functional Mockup Interface) wurden direkt im Projekt aufgegriffen. Auch der Credible Simulation Process hat seine Wurzel in SmartSE.

Damit sind die wesentlichen Teile der fachlichen Aufsetzbasis von SET Level skizziert. Um den Stand der Wissenschaft und Technik umfassend zu beschreiben, müsste eine große Zahl an Quellen referenziert werden, da die Absicherung automatisierter Fahrfunktionen von vielerlei Seiten bearbeitet wurde und wird. An dieser Stelle sei auf die Dokumentationen der oben besprochenen drei Projekte verwiesen, die mit ihren Dokumentationen bereits einen großen Teil davon abdecken.

Im Rahmen dieses Projektes wird PROSTEP eine entsprechende Architektur entwerfen und entsprechende Produkterweiterungen in dem von PROSTEP in das SET Level Projekt eingebrachte „Traceability Software“ durchzuführen, um effektive Traceability zu ermöglichen. Hierbei greift PROSTEP auf die Erfahrung der Entwicklung der Produkte OpenPDM und OpenDXM GlobalX zurück, die am Markt die von über 200 Kunden weltweit eingesetzt werden.

Außerdem hat PROSTEP jahrelange Erfahrung im Bereich des allgemeinen „Systems Engineering“ und insbesondere dem „Model-based Systems Engineering“. Beispiele hierfür sind die durchgängige, fachbereichsübergreifende Modellierung z. B. von Prozessen, sowie die Erstellung von (deskriptiven) Systemmodellen oder die Erstellung von Artefaktmodellen, die während des gesamten Produktentstehungsprozesses eingesetzt werden können.

Im SET Level-Projekt bringt PROSTEP die Systems Engineering, Prozess-, Methoden- und Modellierungskompetenz, die Projektergebnisse aus dem SmartSE-Projekt, sowie speziell die Kompetenz als neutraler Berater ein.

0.5 Zusammenarbeit mit anderen Stellen

Im Rahmen des SET Level Projektes wurde mit dem Schwesterprojekt VVMethoden, dem prostep ivip Gremium SmartSE und der Modelica Association zusammengearbeitet.

Darüber hinaus wurde die prostep-eigene Tochterfirma mit Entwicklungsaufgaben im Umfeld TRACY betraut. Die Ergebnisse sind im Berichtsteil des AP4.1 zusammen mit den von der PROSTEP AG durchgeführten Entwicklungen beschrieben.

1 Eingehende Darstellung

1.1 Erzielte Ergebnisse

1.1.1 Teilprojekt 1: Use-Case Management und funktionsorientierte Anforderungsanalyse

1.1.1.1 AP 1.1 Anforderungen an die Simulation

Das AP 1.1 hat das Ziel die Anforderungen an die Simulation und ihre Implementierungsform zu definieren. Dies soll auch in Absprache mit dem VVMethoden-Projekt geschehen. Zum einen soll dabei sichergestellt werden, dass die Simulation solche funktionalen Leistungen erbringen kann, wie sie in Anwendungskontexten besonders nachgefragt werden. Zum anderen soll eine möglichst hohe Kompatibilität der Realisierung angestrebt werden. Dies betrifft sowohl technische Schnittstellen wie auch Formate zur Repräsentation von Testaufgaben und Testresultaten. Ziel ist es, die Entwicklung einer Simulationstechnologie zu unterstützen, welche die an sie gestellten Aufgaben lösen kann und sich in den Gesamtprozess der Absicherung einer automatisierten Fahrfunktion gut einbetten lässt.

In das AP1.1 hat PROSTEP seine Erfahrungen und Kompetenzen im Kontext Simulation eingebracht. In enger Abstimmung mit TP2 wurden Anforderungstabellen für das die Simulation des hochautomatisierten Fahrens erarbeitet. Zudem hat PROSTEP aktiv Beiträge zur Spezifikation der Use Cases 1 bis 3 geleistet. Zur Sicherstellung einer durchgängigen Informationsqualität im Projekt wurde ein gemeinsamer Reviewprozess für TP1 und das Gesamtprojekt festgelegt. Der initiale Anforderungs- und Reviewprozess ist in Abbildung 6 dargestellt.

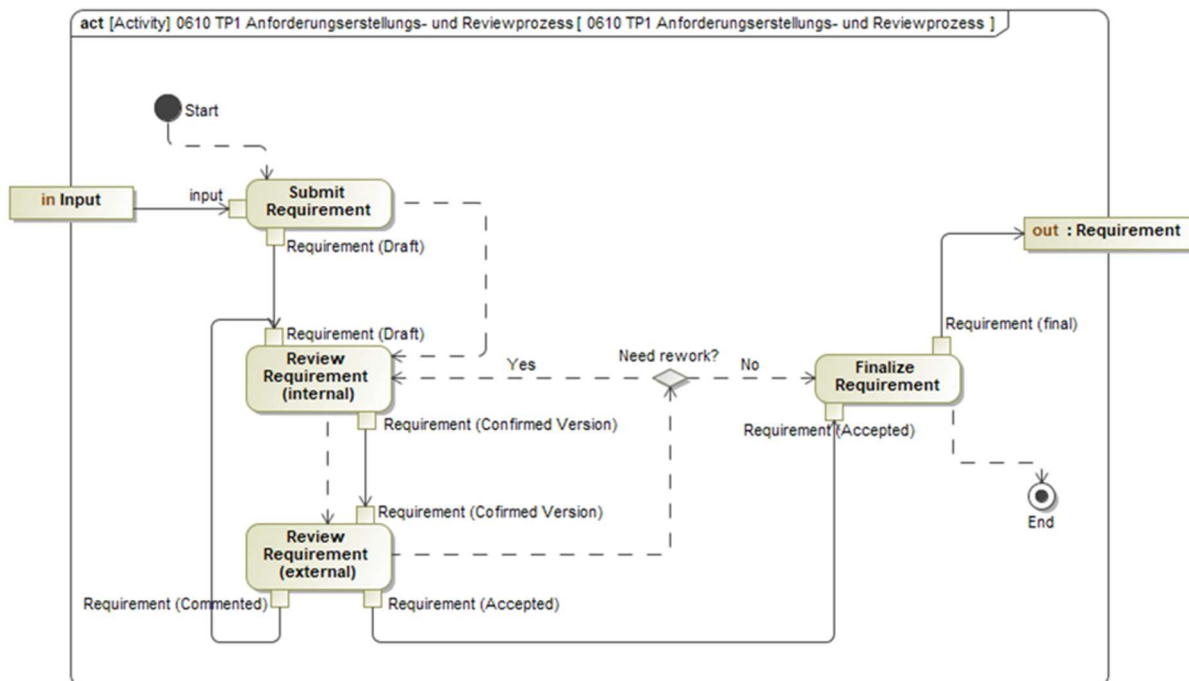


Abbildung 6: Anforderungs- und Reviewprozess

Während der Projektlaufzeit wurden die Anforderungen an die Simulation – funktional und qualitativ in enger Abstimmung mit dem Projekt VVMethoden formuliert, woraufhin PROSTEP den gemeinsamen Reviewprozess hin zu einem Austausch- und Reviewprozess weiterentwickelte, um die Bedürfnisse eines projektinternen - und eines projektexternen Reviews zu berücksichtigen. Der weiterentwickelte Prozess ist in Abbildung 7 dargestellt.

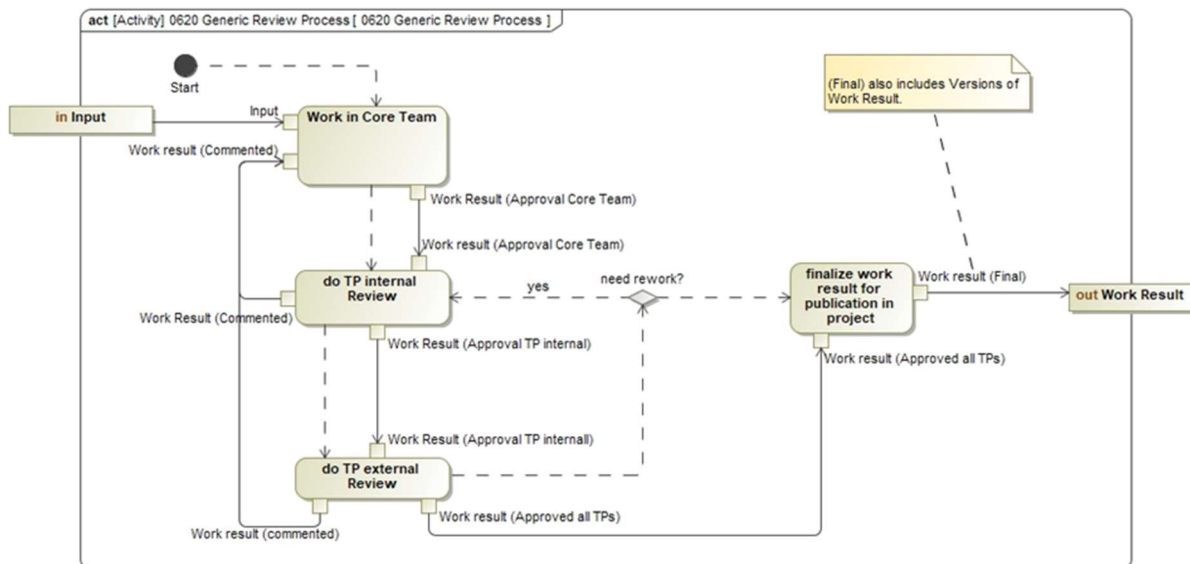


Abbildung 7: Weiterentwickelter Anforderungs- und Reviewprozess

PROSTEP hat hier aktiv zu einem reibungslosen Austausch der Anforderungen mit VVMethoden beigetragen. Durch eine regelmäßige Teilnahme an den internen Review-Meetings sowie durch die weiterhin durchgeführte Mitarbeit der VVMethoden Arbeitspakete, die am Review beteiligt sind, konnte ein reibungsloser Ablauf im Austausch zwischen VVMethoden und SET Level erwirkt werden und weiterer Input für die Anforderungen an die Simulation erarbeitet werden. Im weiteren Projektverlauf wurde in einer von AP 1.1 initiierten Review-Runde, ein erneutes Review bei den VVMethoden-Arbeitspaketen angefragt. Die Kommunikation zu den VVMethoden-Arbeitspaketen hat an dieser Stelle PROSTEP übernommen und fungierte als Schnittstelle zwischen den Projekten.

Zusätzlich zur Kommunikation mit VVMethoden hat PROSTEP sich aktiv an der Fertigstellung der Anforderungen auf Basis des Feedbacks der Simulation Use Cases, der anderen Arbeitspakete und des Feedbacks aus VVMethoden beteiligt.

Darüber hinaus hat PROSTEP aktiv die Anforderungen zum Review an die Forschungsimplementierung des DLR aufgenommen und weitergegeben. Die Anforderungen wurden in das gemeinsame AP1.1 Anforderungs-Dokument übernommen.

AP 1.2 Szenarienräume und Referenzszenarien

Das AP1.2 beschreibt die exemplarischen Anwendungsräume, an denen die Simulationstechniken und Modelle entwickelt und evaluiert werden sollen. Insbesondere gehört dazu der Szenarioraum „Komplexe Kreuzung“, der mit anderen ÖGPs abgestimmt wird. Als Fokus-Anwendungsfälle werden Referenzszenarien ausgewählt, an denen sich die Entwicklungen der anderen Teilprojekte des SET Level Projektes orientieren und die in der internen Bewertung der Arbeitsergebnisse Verwendung finden. In diesen Referenzszenarien sollen die statische Umgebung, Umweltparameter (z. B. Wetter), beteiligte Verkehrsteilnehmer und Fahrmanöver konkretisiert werden.

Für eine adäquate Darstellung der Anwendungsfälle und Referenzszenarien entwickelt das SET Level Projekt geeignete Sprachmittel. Diese stellen einen Teil der umfassenden Sprachmittel für die Beschreibung verschiedener Aspekte von Szenarien dar. Im TP 1 sollen dazu die abstrakteren Sprachebenen festgelegt werden.

Dazu wurden initial die die Anforderungen an die Szenarien wurden auf einen im Szenario-Workshop in Karlsruhe vertieft. Im Arbeitspaket wurden darauf aufbauend verschiedene Szenarien erstellt. Diese wurden nach einem im Arbeitspaket entwickelten Konzept kategorisiert und den Verkehrsräumen zugeordnet. Es wurden die von den Partnern bereitgestellten

Verkehrsräume und abgeleiteten Szenarien untersucht und die Anforderungen an das Beschreibungsformat OpenSCENARIO identifiziert.

Für ein gemeinsames Verständnis wurde zusammen mit den Partnern ein Projektglossar erstellt und die Klärung verschiedener Begrifflichkeiten für das Arbeitspaket organisiert. Ebenfalls wurde eine Kategorisierung der verschiedenen Verkehrsräume in drei Kategorien eingeführt.

Die im Arbeitspaket erstellten Verkehrsräume wurden einem Review unterzogen. Hierbei hat PROSTEP ihre Brückenfunktion wahrgenommen und Erkenntnisse bezüglich des Verkehrsraums aus den anderen Arbeitspaketen eingebracht. Für den ersten Meilenstein wurde in enger Abstimmung mit AP 2.1 das Szenario „Rechtsabbiegen“ für den Verkehrsraum „einfache Kreuzung“ detailliert. Hierzu wurden durch das Arbeitspaket, openDRIVE Dateien für den Verkehrsraum erstellt und die erstellten openSCENARIO Dateien einem kritischen Review unterzogen.

Weiterhin fand eine Befragung der Simulation Use Cases statt, welche Anforderungen an den Inhalt weiterer Szenarien gestellt werden. PROSTEP hat sich aktiv in die Aufnahme und die Priorisierung der Anforderungen der Simulation Use Cases, sowie in die Erweiterung und Ausdetaillierung der durch AP 1.2 erstellten Verkehrsräume und Szenarien eingebracht. Auf dieser Basis wurde die Formulierung weiterer Szenarien durchgeführt. Dabei wurden speziell das Referenzszenario RS202 im Verkehrsraum VR202 (vgl. Abbildung 8) um weitere Verkehrsteilnehmer erweitert, sowie die Referenzszenarien RS204 und RS205, die auf dem Verkehrsraum VR202 beschrieben wurden weiter detailliert. PROSTEP hat sich dabei aktiv am Review beteiligt und sich in der Rolle der Brückenfunktion mit Erkenntnissen aus den anderen APs zur detaillierteren Beschreibung der Referenzszenarien eingebracht.

Reference Scenarios – RS2

Right turn with crossing pedestrians

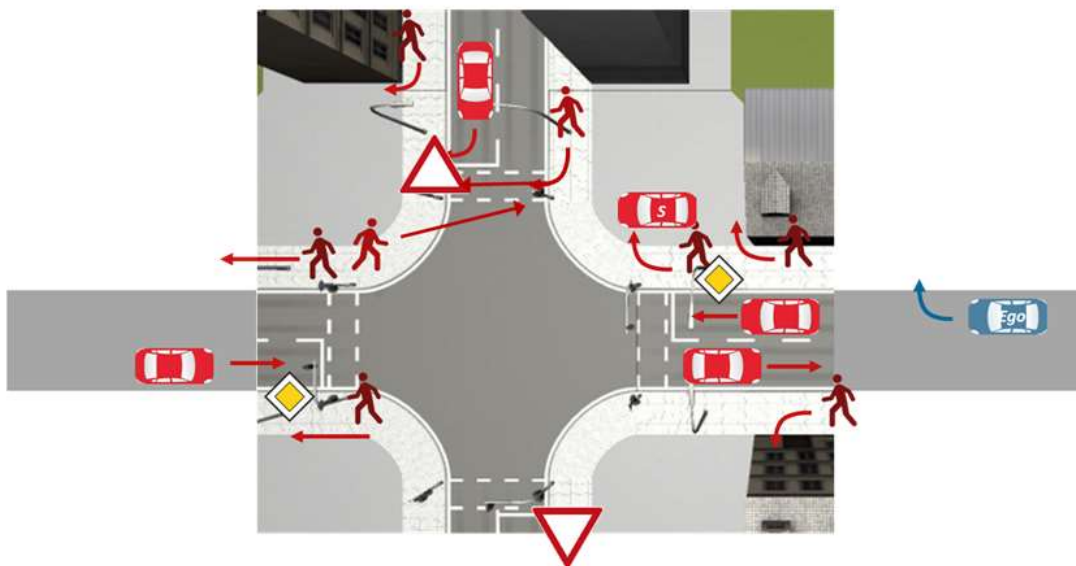


Abbildung 8: Referenzszenario RS202 im Verkehrsraum VR202

Zusätzlich wurde eine Konsolidierung der im AP1.2 entstandenen Referenzszenarien und Verkehrsräume durch das Schwesterprojekt VVMethoden initiiert. In Zuge dessen und im Hinblick auf wurden die definierten Verkehrsräume und Referenzszenarien auf einen konsistenten, zur Weitergabe geeigneten Stand gebracht. PROSTEP hat hierbei aktiv an den Abstimmungsrunden des APs teilgenommen und die Beschreibung der Referenzszenarien wurde durch PROSTEP einem kritischen Review unterzogen und Feedback eingebracht.

Basierend auf diesen Arbeiten erfolgte nach dem Halbzeitevent der Abgleich der Referenzszenarien mit den „Functional Use Cases“ aus dem Projekt VVMethoden. PROSTEP hat hier ihr Wissen über aktuelle veröffentlichte Arbeiten aus VVMethoden zur Erweiterung des

Sechs-Ebenen-Modells aus PEGASUS in das AP eingebracht, um eine einheitliche Basis der betrachteten Szenarien zwischen den Projekten zu garantieren. Abbildung 9 zeigt die Sechs-Ebenen-Darstellung. Hierbei hat PROSTEP durch seine Beteiligung an beiden Projekten eine Schnittstellenfunktion eingenommen.

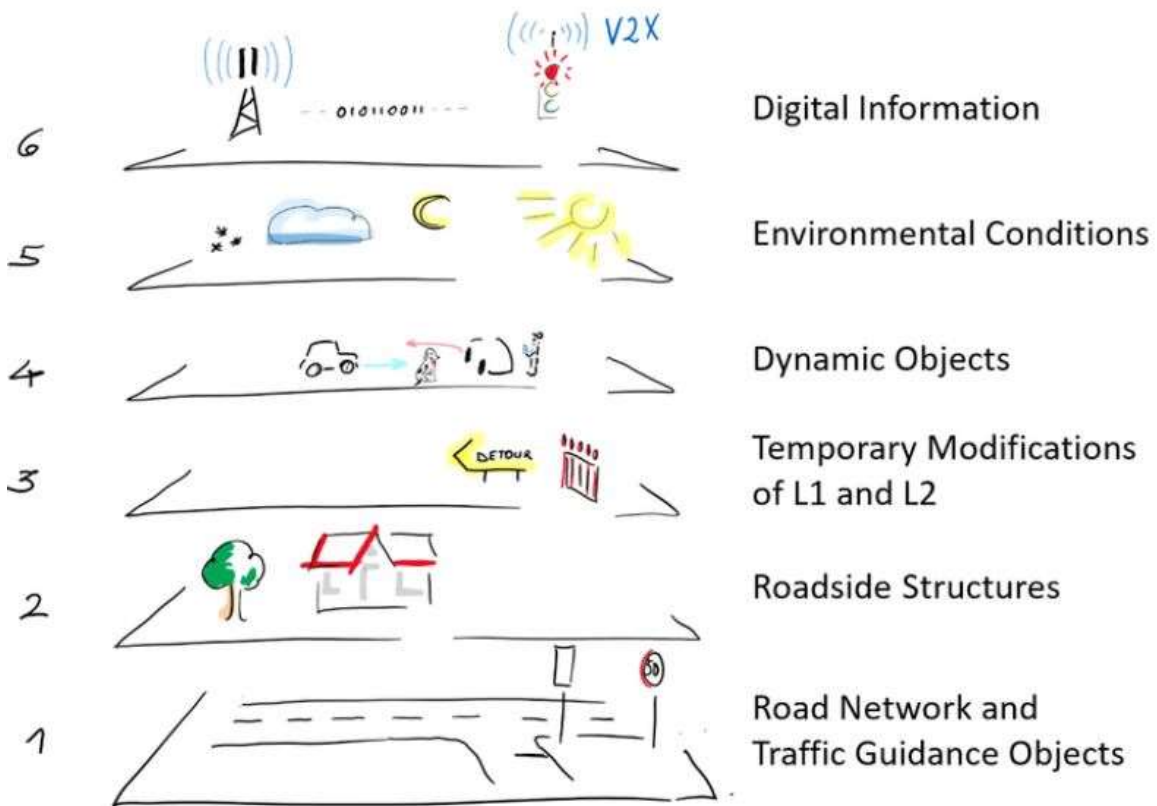


Abbildung 9: 6-Layer Modell [2] als Basis für die Szenarienbeschreibung

1.1.2 Teilprojekt 2: Simulationsbasiertes und virtuelles Entwickeln / Testen

PROSTEP beteiligte sich über die in der VHB vorgesehenen Arbeiten in AP2.1 auch an Arbeiten an weiteren UAPs (z.B. Architektur) des AP2.1, sowie an den Arbeiten zu Standards aus AP2.3. Aufgrund der teilweisen sehr engen Verknüpfung der Arbeiten, werden diese hier gemeinsam dargestellt.

1.1.2.1 AP 2.1 Aufbau einer einheitlichen Integrationsarchitektur

Standards und Szenarienbeschreibung

Bereits im PEGASUS-Projekt wurde für die Szenarienbeschreibung der Standard OpenSCENARIO genutzt. SET Level setzte ebenfalls auf diesem bereits etablierten Standard auf. Im Rahmen der Vorarbeiten zur Prüfung inwieweit der Standard die Anforderungen, die sich aus den Simulationaufgaben für automatisierte Systeme im urbanen Raum ergeben, erfüllt, wurde zur durchgängigen Beschreibung dieses Open-Standards ein SysML-Modell von OpenScenario erstellt und zur Unterstützung der Spezifikation der Szenarien bereitgestellt. Abbildung 10 zeigt den Aufbau dieses Modells.

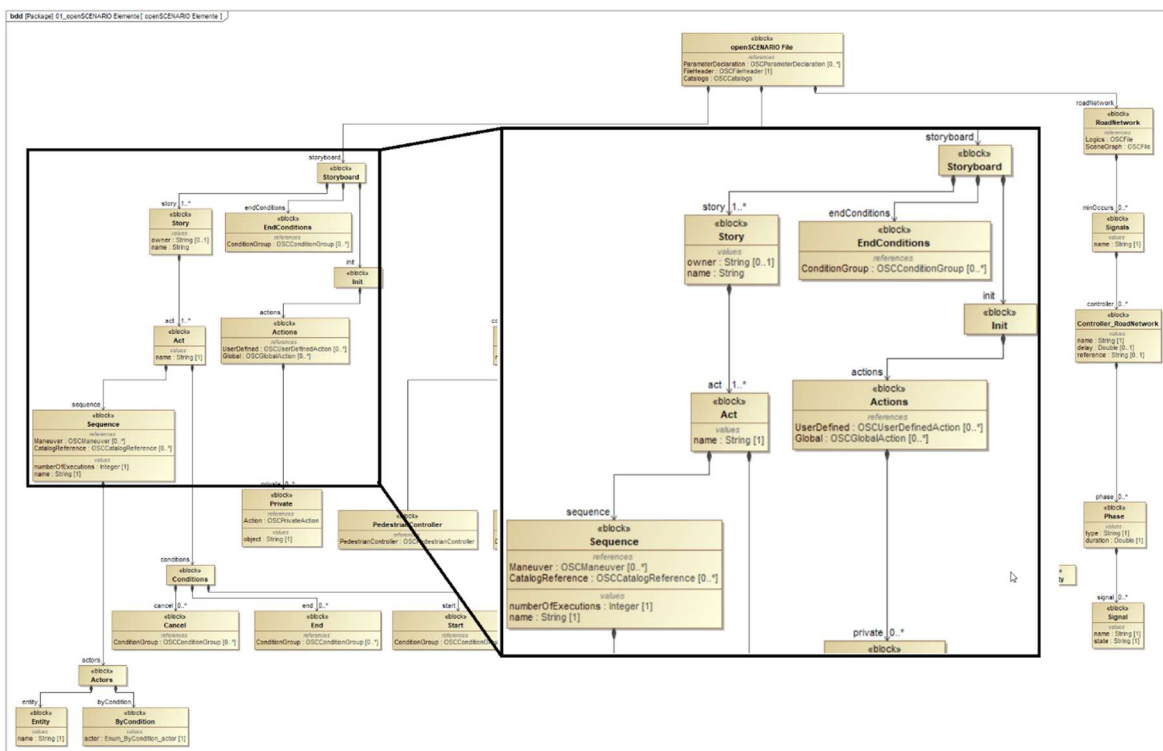


Abbildung 10: SysML-Modell für OpenSCENARIO

Vor dem Hintergrund der Fragestellung, inwieweit der Standard die in SET Level gestellten Anforderungen erfüllt, beteiligte sich PROSTEP aktiv an der Diskussion über die nötigen Erweiterungen von OpenSCENARIO, um zu erörtern, welche Fahrmanöver nötig sind, um die Simulation Use Cases zu realisieren. Darüber hinaus wurde ebenfalls diskutiert, inwieweit OpenSCENARIO erweitert werden kann, um es unter anderem zu ermöglichen, dass Verkehrsteilnehmer Verkehrsregeln ignorieren, um so kritische Situationen zu erzeugen. Der Fokus lag dabei auf der Erweiterung der OSI:TrafficCommands. Hierzu wurden auch Anforderungen aus den Simulation Use Cases über das AP4.1 eingebracht.

PROSTEP beteiligte sich darüber hinaus an den Arbeiten zur Testfallbeschreibung und Testspezifikation. Hierbei wurden auch die Arbeiten zum Credible Simulation Prozess aus dem TP3 mitberücksichtigt. Des Weiteren brachte sich PROSTEP im Rahmen der Testbeschreibung in die Definition der KPIs ein. Hierbei wurde an der Evaluierung der zum MS1 in den SUCs verwendeten KPIs mitgearbeitet und sich an den Diskussionen über eine formelle Beschreibung von KPIs beteiligt. Dazu wurde über die Verbindung mit TP4.1 auch Anforderungen aus den SUCs mit eingebracht.

Für die Klassifizierung weiterer für SET Level relevanter Standards wurde eine Übersicht der den Projektpartnern bekannten und von diesen verwendeten Standards erstellt. Basierend auf dieser Übersicht fand eine Bewertung hinsichtlich Relevanz und Anwendung im Projektkontext statt. Für die Einordnung und Bewertung der Standards und der Entscheidungsfindung wurde eine übersichtliche Darstellung gesucht. PROSTEP hat als Vorschlag ein Radarschaubild eingebracht und gemeinsam mit den Projektpartnern an die Erfordernisse von SET Level angepasst. Abbildung 12 zeigt das dazu verwendete Radarschaubild, in dem die verschiedenen Standards nach Relevanz für SET Level und dem Reifegrad der jeweiligen Standards dargestellt sind. Dazu ist in Abbildung 11 die Einteilung der jeweiligen Quadranten erläutert.

Für die Erfassung und Bewertung der Standards wurde zudem insbesondere die Abstimmung mit den korrespondierenden Aktivitäten im TPX des Projekts VVMethoden und ein Abgleich mit der ISO-Datenbank durchgeführt.

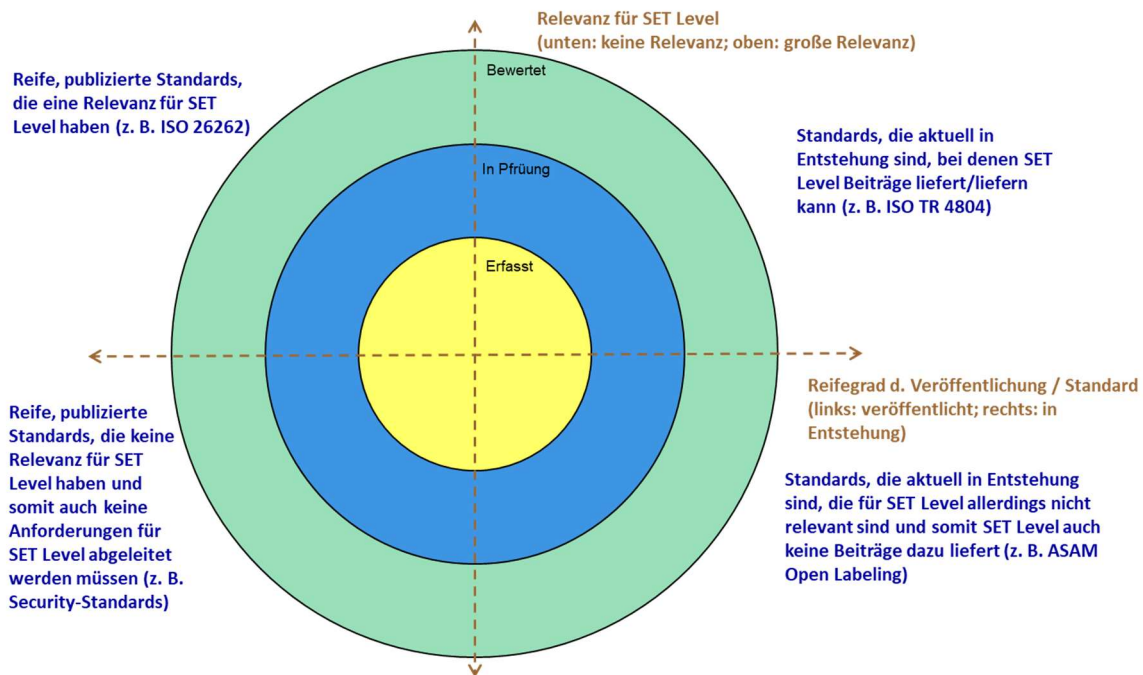


Abbildung 11: Radardiagramm der für SET Level relevanten Standards, Erläuterungen zu Quadranten

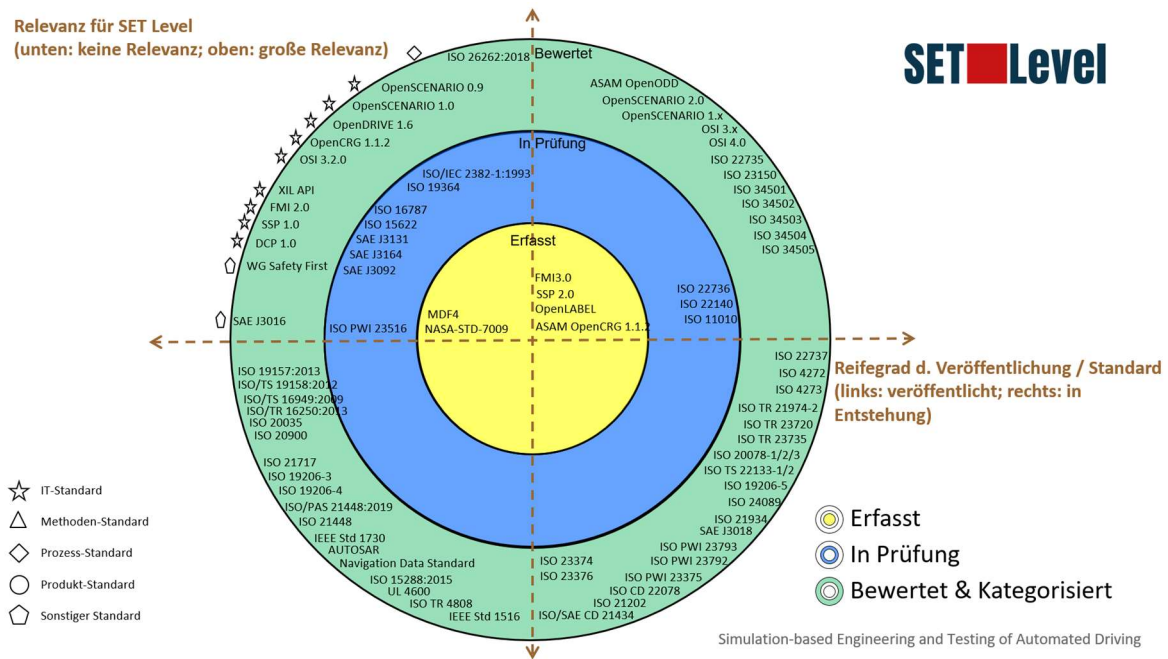


Abbildung 12: Beispiel für das Radardiagramm der für SET Level relevanten Standards, eingeteilt nach Relevanz und Reifegrad

Architektur

Eine hauptsächliche Motivation für das SET Level Projekt beruht auf der Erkenntnis, dass hochautomatisierte Systeme nicht mehr allein durch physikalische Tests abgesichert werden können. Für die Absicherung der zugrunde liegenden Funktionen werden verstärkt Simulationen zum Einsatz kommen.

Ein Simulationsarchitektur, die in der Lage ist, eine große Bandbreite von Inhalten zur Absicherung angefangen bei Sensormodellen über komplette Verkehrsszenarien bis hin zu Prozessen und den dabei eingesetzten Werkzeugen integrativ in einer Architektur abzubilden, ermöglicht es, die aus der simulationsbasierten Absicherung entstehenden technischen und organisatorischen Herausforderungen zu begegnen.

Im Rahmen des SET Level Projektes wurde, dazu eine integrative Simulationsarchitektur entwickelt, die auf den gängigen Methoden des Model Based Systems Engineering (MBSE) basiert. Sie ermöglicht, die durchgängige Beschreibung von einer abstrakten Systemarchitektur bis hin zu implementierbaren Simulationsmodellprototypen. Die Grundlage dieser Architektur des Gesamtsystems der Simulation bildet dabei der RFLP-Ansatz. Das Gesamtsystem beinhaltet nicht nur das eigentliche Simulationsmodell, sondern auch weitere Artefakte, wie den Simulator oder den der Absicherung bzw. Simulationsaufgabe zugrunde liegenden glaubwürdigen Simulationsprozess.

Für die konkrete Beschreibung der Architektur wurde dabei die Modellierungssprache UML (Unified Modeling Language) verwendet. Damit soll einerseits die Verständlichkeit verbessert werden, andererseits wird durch den Einsatz dieser standardisierten Sprache eine toolunabhängige Lösung forciert.

Im Rahmen der Architekturdiskussion und deren Erstellung wurde dabei intensiv projektübergreifend mit dem AP4.1 zusammengearbeitet. Dabei wurde an der Beschreibung der Schnittstellenarchitektur sowie der Konsolidierung verschiedener im Projekt entstehenden Architekturen mitgearbeitet. Dazu wurden in enger Zusammenarbeit mit dem AP4.1 verschiedenen Architekturbegriffe harmonisiert und die Einführung eines Modellierungswerkzeugs (Sparx Systems' Enterprise Architect) zur Gewährleistung einheitlicher Architekturen im Projekt unterstützt. Dazu wurde darüber hinaus ein Modellierungshandbuch von PROSTEP erstellt, um die einheitliche Modellierung zu gewährleisten.

Title: Workflow: Architecture to SSP and FMU
Types: BPMN 2.0 Collaboration Diagram
 See https://en.wikipedia.org/wiki/Business_Process_Model_and_Notation
Meaning: Deriving the architecture of a specific simulation (task) involves several levels of abstraction, processes, tools, business objects/artifacts and their formats. For better overview, not all message flows are depicted, her. See [Message Flow Diagram](#) for those details.

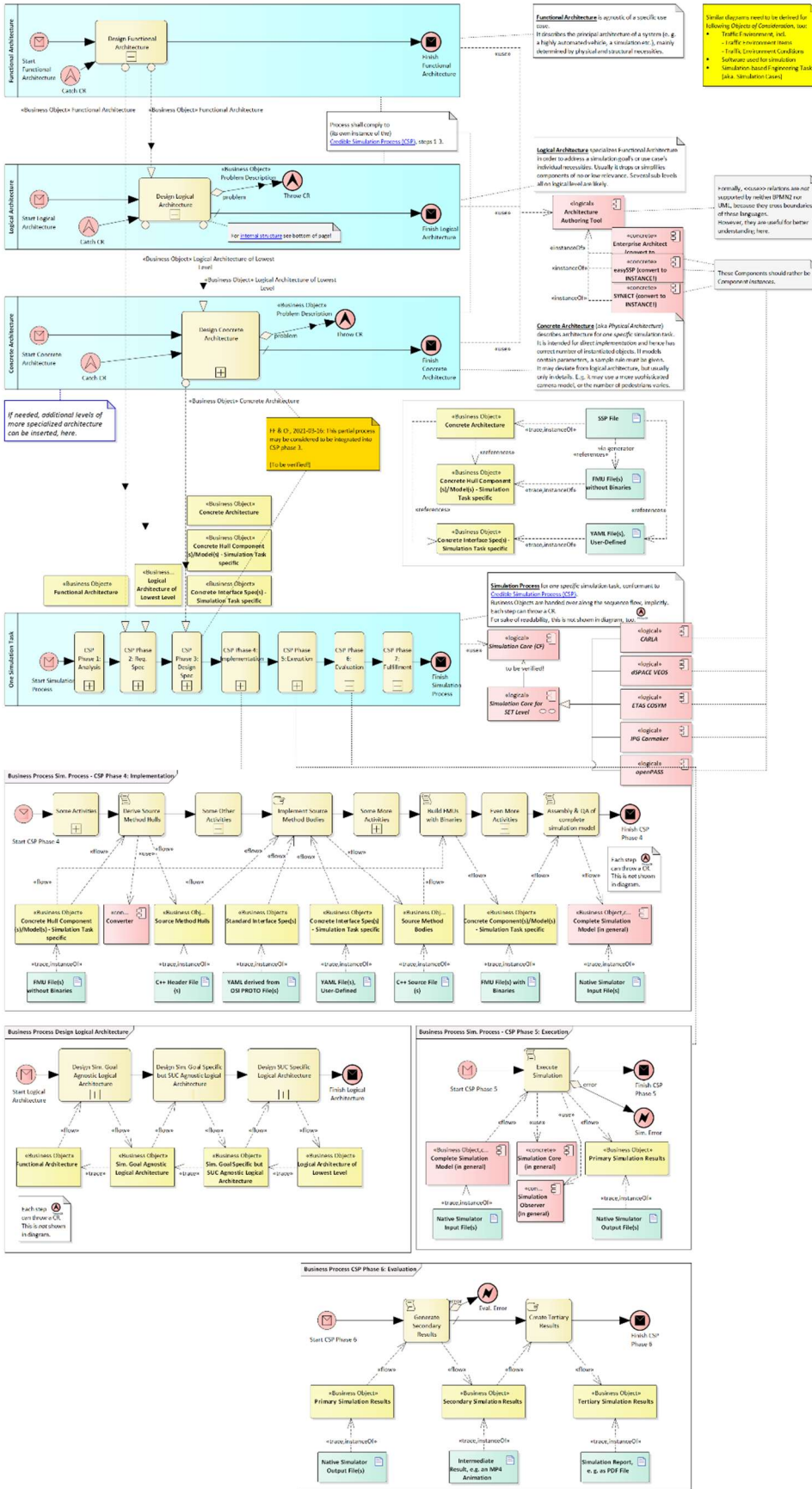


Abbildung 13: Exemplarische Verknüpfung der drei Architekturebenen mit dem Credible Simulation Prozess

In enger Zusammenarbeit mit AP 4.1 wurden drei Ebenen einer Simulationsarchitektur definiert und mit dem Credible Simulation Process aus TP 3 verknüpft. Die Verknüpfung dieser Ebene mit dem Credible Simulation Process ist exemplarisch in Abbildung 13 dargestellt. Darüber hinaus wurde zusammen mit dem AP4.1 ein „Proof of Concept“ (PoC) entwickelt, wie sich ein durchgängiger toolgestützter Prozess von den höchsten, auf funktionaler Ebene abstrahierten Modellarchitekturen bis hin zu Source-Code-Rümpfen für z. B. Sensor-Modelle darstellen lässt. Auch dies ist in Abbildung 13 mit dargestellt.

Parallel zu diesen theoretischen Arbeiten wurde die praktische Umsetzung der diskutierten Architekturelemente in Enterprise Architekt (EA) gesteuert und mitgeprägt. Dazu wurde in enger Zusammenarbeit mit ZF ein Konzept zur automatisierten Generierung von System Structure and Parameterization (SSP) aus einem EA-Export erprobt. Dafür wurden unter anderem Regeln definiert, wie Ports und Interfaces zu benennen und zu verbinden sind. Abbildung 14 zeigt exemplarisch die Überführung eines Architekturelements aus dem Enterprise Architekt in ein standardisiertes SSP. Damit kann dieses Element auch in anderen Tools genutzt werden.

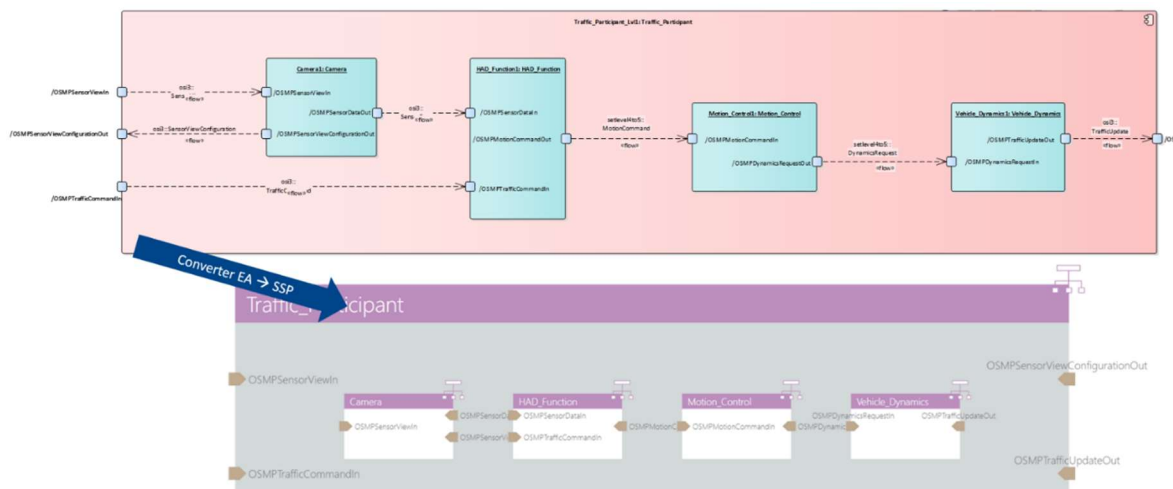


Abbildung 14: Übertragung von Architektur nach SSP und weiter in andere Tools

Prozesssichere Gesamtbeschreibung

Die Arbeiten zur prozesssicheren Gesamtbeschreibung wurden von PROSTEP geleitet. Um eine prozesssichere Gesamtbeschreibung einer Simulationsaufgabe zu etablieren, wurden zunächst die Arbeiten in den Simulation Use Cases bis Meilenstein 1 retrospektiv betrachtet. Dazu wurde ein Konzept entwickelt, wie der Stand der SUCs erhoben werden kann und daraufhin Interviews mit den SUC-Verantwortlichen durchgeführt. Im Zuge dessen wurden alle verfügbaren Informationen zusammengetragen und dokumentiert. In Abstimmung mit den Beteiligten wurde daraufhin beschlossen, die Dokumentation in GitLab durchzuführen.

Diese Dokumentation wurde anschließend auf den in TP 3 erarbeiteten Credible Simulation Prozess (CSP) gemappt. Zum einen, um den CSP zu evaluieren und zum anderen, um die Dokumentation ab Meilenstein 2 effizienter gestalten zu können. Abbildung 15 zeigt exemplarisch einen Ausschnitt der Dokumentation in GitLab. Eine wesentliche Erkenntnis daraus war, dass zwar „unterbewusst“ nach dem Prozess gearbeitet wurde, aber die Dokumentation noch nicht nach dem Prozess durchgeführt wurde, da dieser auch noch im Entstehen war. Das Wissen war implizit bei den beteiligten Personen vorhanden und wurde daher auch so angewandt, was zeigt, dass der CSP gängige Ingenieurspraxis widerspiegelt.

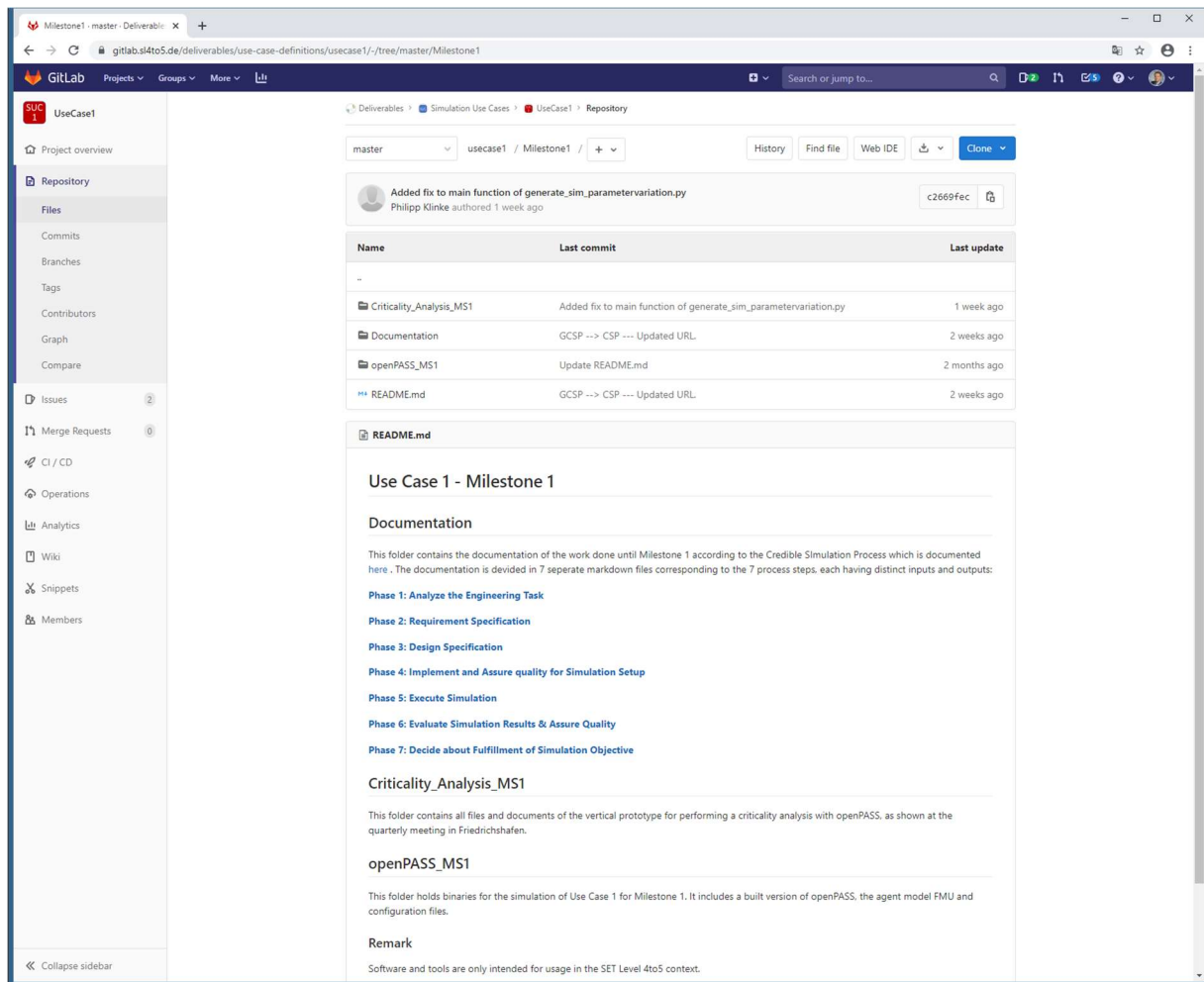


Abbildung 15: Übersicht SUC1 Dokumentation in GitLab

Zum Start von Meilenstein 2 wurden auf Basis einer zuvor erstellten Dokumentation, Prozess-templates für die Git-Repos aller SUCs erstellt. Darin wird explizit aufgeführt, was Inhalt/Input/Output der einzelnen Prozessschritte ist (siehe Abbildung 16). Anschließend wurden sie den SUCs an die Hand gegeben und dienen als Prozess- und Dokumentationsvorlage. Des Weiteren hat AP 2.1 die Dokumentation und die Arbeit mit dem Prozess aktiv unterstützt. Letztlich dienen die dokumentierten Artefakte auch als Input für TRACY in AP 4.1.

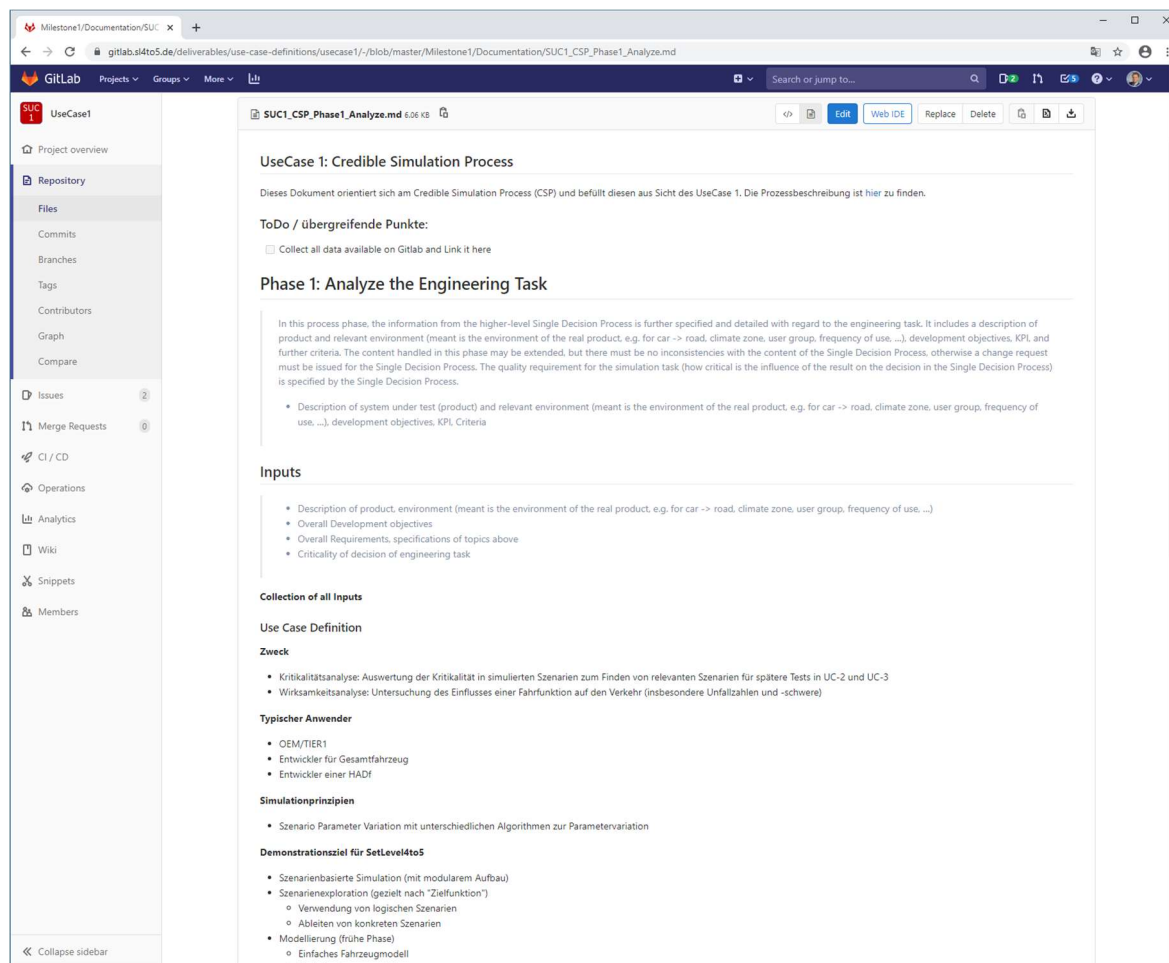


Abbildung 16: Phase 1 SUC1 Dokumentation in GitLab

Die erstellten Templates für die Dokumentation der Simulation Use Cases (SUC) auf Basis des Credible Simulation Process (CSP) ermöglichten die einheitliche und strukturierte Ablage. Basierend auf dieser umfangreichen Dokumentation wurde die Struktur in den Traceability-Demonstrator TRACY übertragen. Dieses Vorgehen der Dokumentation nach dem CSP liefert gleichzeitig einen wichtigen Beitrag zum „Proof of Concept“ in AP5.1 und im Speziellen zur dortigen kritischen Reflexion.

Um den allgemein gehaltenen CSP im Projekt besser anzuwenden zu können und die Dokumentation weiter zu vereinfachen, wurde in Zusammenarbeit mit AP 3.1 außerdem eine Checkliste erstellt. Erreicht wird dieses Ziel durch die Formulierung konkreter Fragen bezüglich der Vorgehensweise während des Simulationsprozesses. Die Antworten spiegeln direkt die zu dokumentierenden Artefakte wider.

Die erfassten und dokumentierten Artefakte reichen dabei von Anforderungen über daraus abgeleitete Designspezifikationen bis hin zu den ersten Modellen, welche zum Einsatz kommen sollen. Abbildung 17 zeigt exemplarisch einen Teil der erfassten Modellanforderungen, sowie die daraus resultierenden Spezifikationen für den SUC1 in TRACY. Zusätzlich ist links im Bild (STRUCTURE VIEW) die Zuordnung zum, als Template in TRACY hinterlegten CSP, zu sehen.

The screenshot shows the TRACY web interface for 'DesignModelSpecification'. The left sidebar displays a 'STRUCTURE VIEW' tree with the following structure:

- Criticality Analysis MS2
 - Credible Simulation Process
 - AnalysisPhase
 - AnalyzeEngineeringTask
 - VerifyEngineeringTaskAnalysis
 - RequirementPhase
 - DeriveModelRequirements
 - DeriveParameterRequirements
 - DeriveSimulationEnvironmentRequirements
 - DeriveSimulationIntegrationRequirements
 - DeriveTestCaseRequirements
 - DeriveProcessQualityRequirements
 - VerifyRequirements
 - DesignPhase (highlighted)
 - DesignModelSpecification** (highlighted)
 - DesignParameterSpecification
 - DesignSimulationEnvironmentSpecification
 - DesignSimulationIntegrationSpecification
 - DesignTestCaseSpecification
 - DesignProcessQualitySpecification
 - VerifyDesignSpecification
 - ImplementationPhase
 - ImplementModel
 - ImplementParameter
 - ImplementSimulationEnvironment
 - ImplementTestCase
 - IntegrateSimulation
 - AssureSimulationSetupQuality
 - DeriveSimulationSetupQualityVerdict
 - ExecutionPhase
 - EvaluationPhase
 - FulfillmentPhase

The main content area shows the 'DesignModelSpecification' form with the following sections:

- Basics**
- Sub-Activities / Components**
- Further Information**
- Metadata**
- Inputs**
 - Add new entry with

<input type="checkbox"/>	Title	Type	Last Editor	Last Modification	Relation	Impact Analysis
<input type="checkbox"/>	SUC1-FM01	requirement	pqm	10.12.2020 23:39	is input for	Start analysis
<input type="checkbox"/>	SUC1-FM02	requirement	pqm	10.12.2020 23:39	is input for	Start analysis
<input type="checkbox"/>	SUC1-FM03	requirement	pqm	10.12.2020 23:39	is input for	Start analysis
<input type="checkbox"/>	SUC1-FM04	requirement	pqm	10.12.2020 23:39	is input for	Start analysis
<input type="checkbox"/>	SUC1-FM05	requirement	pqm	10.12.2020 23:39	is input for	Start analysis
- Requirements**
- Others**
- Outputs**
 - Add new entry with

<input type="checkbox"/>	Title	Type	Last Editor	Last Modification	Relation	Impact Analysis
<input type="checkbox"/>	SUC1-Spec-VSM	design specification	pqm	10.12.2020 23:39	is output of	Start analysis
<input type="checkbox"/>	SUC1-Spec-OSIPed	design specification	pqm	10.12.2020 23:39	is output of	Start analysis
<input type="checkbox"/>	SUC1-Spec-FM	design specification	pqm	10.12.2020 23:39	is output of	Start analysis
<input type="checkbox"/>	SUC1-Spec-VDM	design specification	pqm	10.12.2020 23:39	is output of	Start analysis
- Rationale:**

Abbildung 17: Modellanforderungen SUC1 in TRACY

In Abbildung 18 ist beispielhaft die Repräsentation des Fußgängermodells „OsiPedestrian“ in TRACY zu sehen. Im oberen Teil des Bildes ist die Verknüpfung ins GitLab zu erkennen. Mittig ist eine Beschreibung des Artefaktes hinterlegt und im unteren Bereich ist der „Trace-Graph“ abgebildet, welcher die Verknüpfung des OsiPedestrians mit anderen Artefakten und Prozessschritten im Credible Simulation Process abbildet.

Zu Projektende hin wurde die Dokumentation der SUCs basierend auf der von den SUCs in GitLab zur Verfügung gestellten Dokumentation in TRACY überarbeitet, um dem in AP4.1 neu entwickelten Datenmodell für TRACY Rechnung zu tragen. Die Durchführung erfolgte hierbei in enger Zusammenarbeit mit den AP5.1, um den „Proof of Concept“ zu unterstützen. Damit konnten die Simulation Use Cases basierend auf den erstellten Checklisten zum CSP im, in AP4.1 entwickelten Traceability Demonstrator, konsistent für unterschiedliche Meilensteine beschrieben werden. Das neue Datenmodell ermöglicht zudem die Erstellung von konkreten Baselines zu den einzelnen Meilensteinen. Die in TRACY durchgeführte Dokumentation wurde in XML und zip ausgeleitet und auf GitLab für die jeweiligen SUCs abgelegt, um sie den Projektpartnern auch nach Projektende, ohne die Notwendigkeit eines Zugriffs auf TRACY zur Verfügung zu stellen.

The screenshot displays the TRACY web interface for the 'OsiPedestrian' artifact. The browser address bar shows the URL: `selrx02.pсад.de/tracy/artifact-view/5fd2a3b4c56a7d63b06c0e55`. The page title is 'Tracy - Artifact View'.

External Resource:

- Source Repository: Git
- Name: Fußgängermodell
- Download link: <https://github.com/s4t5.de/deliverables/model/traffic-agents/osipedestrian>
- Show link: <https://github.com/s4t5.de/deliverables/model/traffic-agents/osipedestrian>

Attribute	Value	Type
model.name	OsiPedestrian	text

Basics:

- Qualifier: model
- Release State: Published
- Update Status: OPEN
- Creator: Robert Richmond
- Editor: Jane P. Miller
- Creation Date: 10.12.2020 23:39:48
- Last Update: 11.12.2020 11:24:26

Description: This project implements a pedestrian model. The pedestrian finds a suitable path itself.

Current Status:

- The pedestrian searches its own path, based on a grid.
- Multiple pedestrians can be simulated as part of one FMU, following the OSMP specification.
- It reacts to cars.

Limitations:

- The low-level path planning is not implemented yet. Consequence: the pedestrian currently walks from grid-point to grid-point, making the movement quite jerky.
- Since sidewalks are not visible as a separate lane type, the model will not correctly choose sidewalks.

Tags: ImplementationPhase, Input, Published

Graph:

Filters: 6 entity types selected, 11 qualifiers selected, 22 relationship types selected. CLEAR, APPLY

Depth level: 1. Show max. depth: [toggle]

The graph shows a dependency diagram. At the top is the 'OsiPedestrian Artifact - model'. Below it are two activity nodes: 'IntegrateSimulation ACTIVITY' and 'ImplementModel ACTIVITY'. Red arrows indicate dependencies from various input nodes (represented by circles) to these two activities.

Abbildung 18: Fußgängermodell als Artefakt in TRACY

1.1.3 Teilprojekt 3: Modellspezifikation, -Entwicklung und -Validierung

1.1.3.1 AP 3.1 Modellspezifikation und -aufbau

Für die Absicherung von hochautomatisierten Fahrzeugen sind neben rein physikalischen Tests in hohem Maße simulationsbasierte Test- und Absicherungsmethoden notwendig. Eine Grundlage für dieses simulationsbasierte Absichern und Testen sind neben den eigentlichen Simulationen und den dafür notwendigen Modellen, verlässliche Prozesse zur Erstellung und Durchführung der Simulationen. Die zugrunde liegenden Prozesse sollen dabei nicht nur die Erstellung und Durchführung der Simulationen unterstützen, sondern vor allem auch die Durchgängigkeit und Nachvollziehbarkeit bzw. Traceability der einzelnen Prozessschritte und erstellten Artefakte für die Absicherung sicherstellen.

Im Rahmen des SET Level Projektes wurde dafür ein Prozessframework definiert. Es umfasst unter anderem die beiden selbstähnlichen Prozesse, Credible Simulation und Credible Modelling Process, sowie den Vereinbarungsprozess und das Simulation bzw. Modelling Request. Spezieller Wert wurde dabei auch auf die Nachvollziehbarkeit und Traceability mit Fokus auf eine partnerübergreifende Zusammenarbeit gelegt.

Der Credible Simulation Process (CSP) beschreibt die Prozesskette für die Durchführung einer Simulationsaufgabe, von der Analyse der Aufgabe über die Umsetzung bis zur Validierung der Ergebnisse. Initial wurde für die Beschreibung der einzelnen Schritte und Übergabeartefakte des Credible Simulation Process von PROSTEP eine Darstellung in SysML erarbeitet und zur Verfügung gestellt. Eine Ausleitung der Top-Level-Ansicht des zu Grunde liegenden Aktivitäten-Diagramms ist in Abbildung 20 abgebildet.

Für die Anwendung des Credible Simulation Process innerhalb des SET Level Projektes wurde dieser in einer Markdown-Datei direkt in GitLab beschrieben und abgelegt. Dabei fand eine Beteiligung sowohl an der konkreten Definition und Ausformulierung des Prozesses sowie der einheitlichen Ablage in GitLab statt. Zudem flossen in die Definition des Prozesses Erkenntnisse aus dem AP 5.2 ein. Eine Darstellung des finalen Credible Simulation Process ist in Abbildung 19 dargestellt.

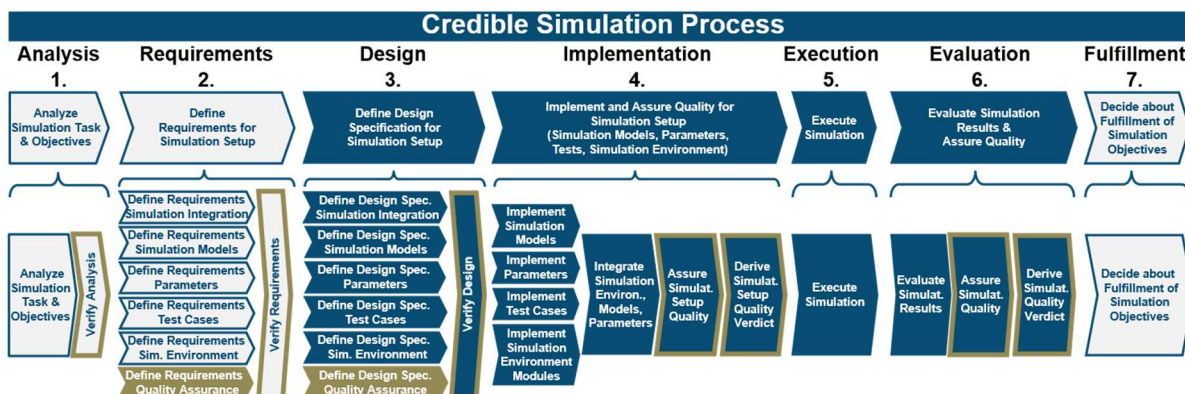


Abbildung 19: Aufbau des Credible Simulation Process

Bei der Ausdetaillierung des Prozesses wurde speziell darauf geachtet, dass der Credible Simulation Process kompatibel zu dem höherliegenden Prozess ist. Dies betrifft die ersten beiden Phasen des CSP „Analyze Simulation Task & Objectives“ und „Define Requirements“ und die letzte Phase „Decide about Fulfillment of Simulation Objectives“. Dabei soll sichergestellt werden, dass eine mögliche Angliederung an den auf dem nächsthöheren Level liegenden Prozess „Simulation-based Engineering Task“ möglich ist und trotzdem zu gewährleistende Selbstständigkeit des Credible Simulation Process nicht beeinträchtigt wird. Hierzu wurde speziell auf vom Credible Simulation Process benötigte Inputs, wie die Vorgabe einer Simulationsaufgabe, die Beschreibung der zu prüfenden Kriterien und eine Einbettung in eine Ingenieursaufgabe, sowie die für den für die höherliegenden Prozess wichtigen Outputs geachtet. Damit ist ein reibungsloser Austausch zwischen den Prozessen möglich.

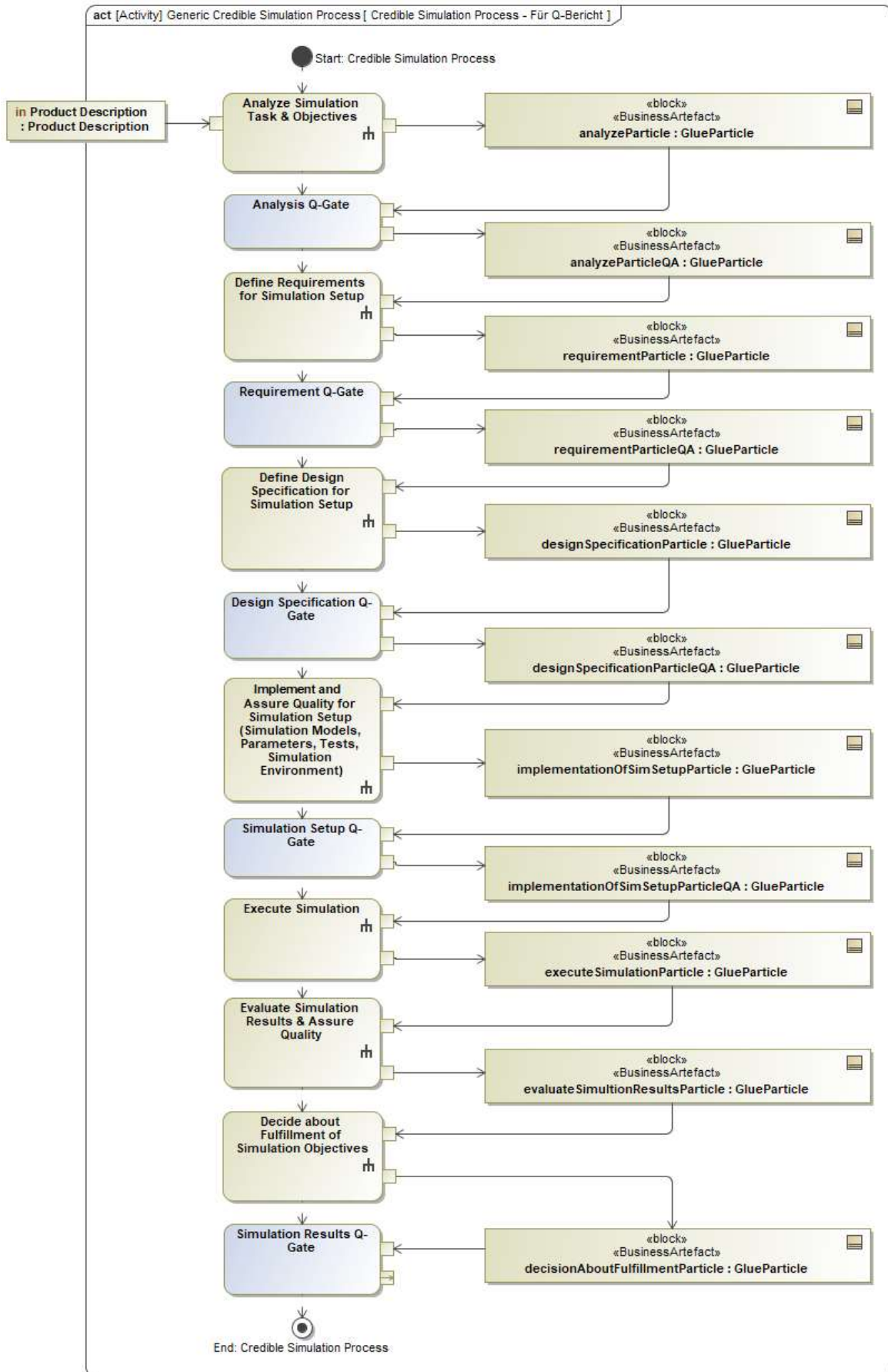


Abbildung 20: Aktivitätsdiagramm des Credible Simulation Process

Die oben angesprochene Beschreibung des Credible Simulation Process in Form einer für das gesamte Projekt zugänglichen Markdown-Datei im GitLab verbesserte einerseits die Möglichkeiten zu einem TP-übergreifenden Review der einzelnen Firmen und ist andererseits auch für die Anwendung in den Simulation Use Cases besser geeignet, da einfacher auf einzelne Textstellen verwiesen werden kann und Feedback kontinuierlich und revisionssicher eingepflegt werden kann. Abbildung 21 zeigt einen Ausschnitt des über GitLab-Issues gesteuerten Feedback- und Anforderungserhebungsprozesses. Dabei wurden beispielweise im Credible Simulation Process verwendete Begriffe genauer geklärt und die Ergebnisse festgehalten.

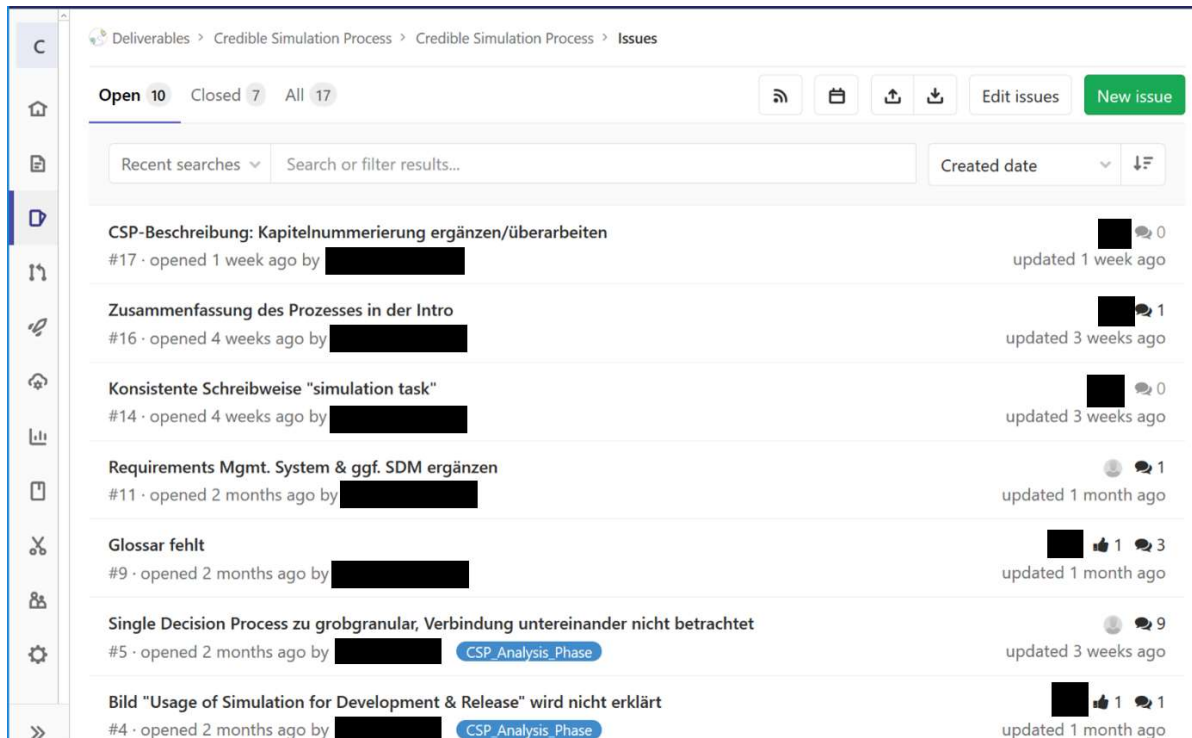


Abbildung 21: Feedback des Credible Simulation Process über GitLab Issues

Für die Anwendung in den Simulation Use Cases wurde neben der Markdown-Datei darüber hinaus in enger Zusammenarbeit mit dem AP 2.1 eine Checkliste für die konkrete Anwendung des Prozesses erstellt. Weiterhin wurde die finale Veröffentlichung des Credible Simulation Process unterstützt und im Zuge dessen zusätzlich auch ein konkretes Beispiel auf Grundlage eines Elektromotors für die Anwendung des Credible Simulation Process mit veröffentlicht. Die einzelnen Schritte des Credible Simulation Process werden beim Durchlaufen des Beispiels mit immer mehr Informationen befüllt: Eine Simulationsaufgabe wird gestellt, darauf aufbauend werden Anforderungen abgeleitet, die durch Designentscheidungen erfüllt werden sollen, usw. am Ende werden die Ergebnisse gegen die Simulationsziele geprüft.

In enger Zusammenarbeit mit dem AP 4.1 und dem AP 2.1 wurden die dort definierten drei Ebenen der Architektur mit dem Credible Simulation Process verknüpft, um die Durchgängigkeit zwischen der Architektur und den im Projekt verwendeten Prozessen zu gewährleisten. Abbildung 22 zeigt exemplarisch die Top-Level-Verknüpfung verschiedener Architekturebenen mit den Phasen des Credible Simulation Process

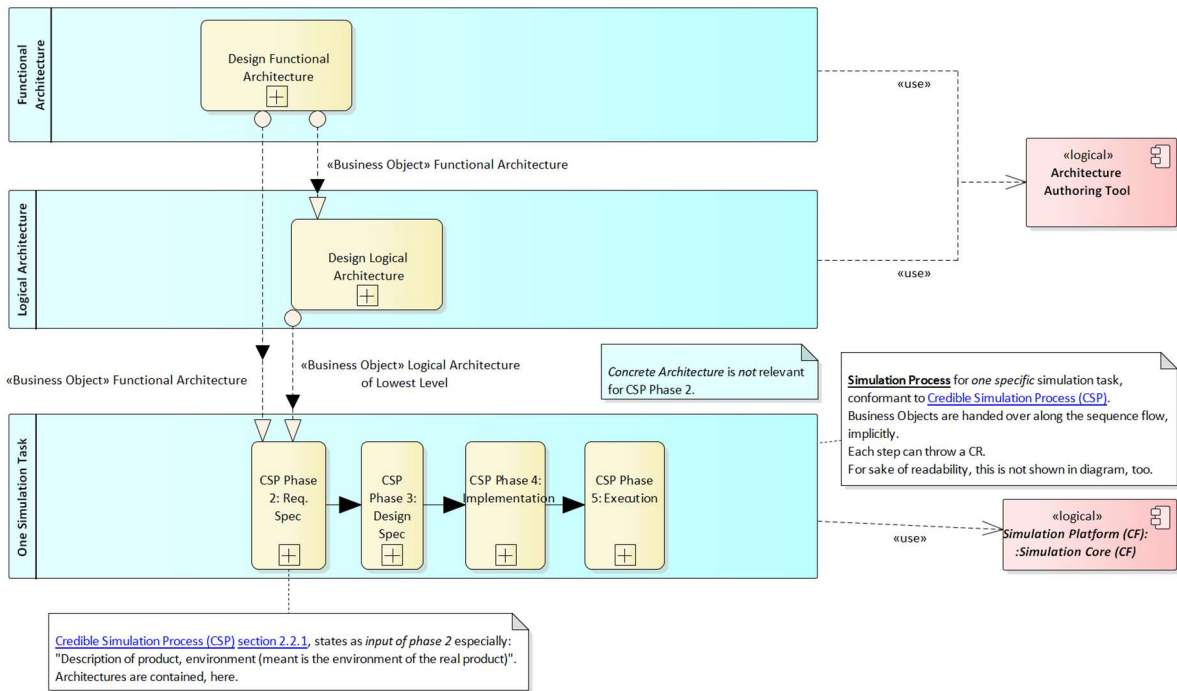


Abbildung 22: Exemplarische Top-Level-Verknüpfung verschiedener Architekturebenen mit den Phasen des Credible Simulation Process

Zusätzlich zum Credible Simulation Process wurde im SET Level Projekt ein dazu selbständiger Prozess, der Credible Modeling Process (CMP) entwickelt. Dieser Modellerstellungs-Prozess beschreibt das Bearbeiten einer Modellierungsaufgabe von der Analyse über das Erstellen bis zum Validieren eines Modells.

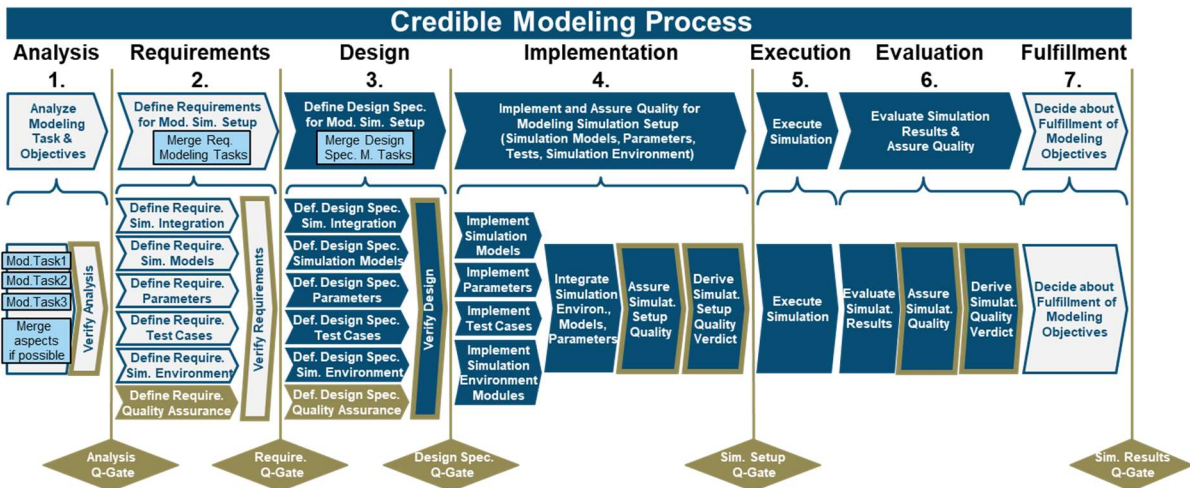


Abbildung 23: Aufbau des Credible Modeling Process

Auch hier wurde die Definition, Erstellung, Dokumentation des Credible Modeling Process unterstützt. Hierzu wurden ebenfalls Erkenntnisse aus den Arbeiten in AP 5.2 mit eingebracht. Die Dokumentation zur Projektlaufzeit fand ebenfalls wie beim Credible Simulation Process in Form einer dem gesamten Projekt zugänglichen Markdown-Dateien direkt in GitLab statt. Der Credible Modeling Process ist, wie in Abbildung 24 als ein Sub-Prozess des Credible Simulation Process definiert.

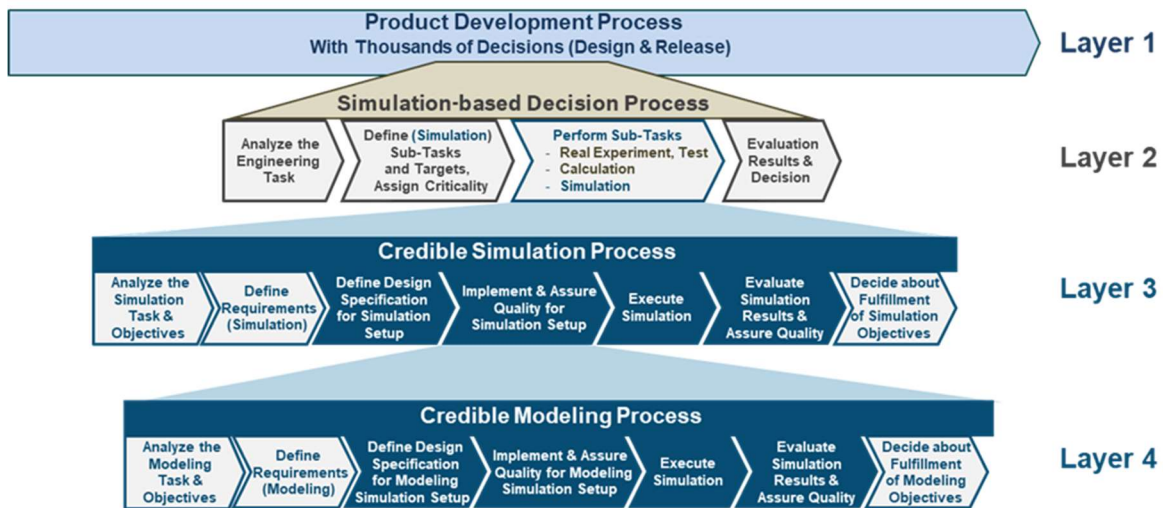


Abbildung 24: Zusammenhang zwischen Simulation-based Decision Process, Credible Simulation und Credible Modelling Process

Bei der Definition des Credible Modeling Process wurde deshalb insbesondere darauf geachtet, dass die Schnittstellen zum Credible Simulation Process eindeutig definiert sind, um eine durchgängige Eingliederung in den Credible Simulation Process zu gewährleisten, aber auch die nötige Abgrenzung zwischen den beiden Prozessen deutlich zu machen. Abbildung 23 zeigt den finalen Stand des Credible Modeling Process.

Für den Credible Modeling Process wurde ebenfalls die finale Veröffentlichung unterstützt und im Zuge dessen zusätzlich auch ein konkretes Beispiel für die Anwendung des Credible Modeling Process auf dem öffentlich zugänglichen Teil des SET Level GitLab mit veröffentlicht.

Innerhalb des SET Level Projektes wurde der Credible Simulation Process für die durchgängige Dokumentation der Simulation Use Cases (SUCs) angewendet. Die Anwendung in den Simulation Use Cases wurde zusammen mit dem AP 2.1 geplant und die betreffenden Partner bei der innerhalb der Simulation Use Cases stattfindenden Dokumentation unterstützt. Dazu wurden die Phasen und Schritte des Credible Simulation Process vor der Anwendung auf die Nutzbarkeit, für die in den Simulation Use Cases beschriebenen Simulationaufgaben überprüft und soweit nötig angepasst. Die gewonnenen Erkenntnisse wurden in die Gesamtbeschreibung des Credible Simulation Process integriert.

Um eine vereinfachte Nutzung des Credible Simulation Process für die durchzuführende Dokumentation innerhalb der Simulation Use Cases zu ermöglichen, wurde darüber hinaus, zusammen mit dem AP 2.1, eine Checkliste für den Credible Simulation Process erstellt. Die Checkliste ist eine Vereinfachung des Credible Simulation Process, der in seiner Originalausprägung allgemeiner gehalten ist, als in den Simulation Use Cases benötigt wird. Die Vereinfachung spiegelt sich in die Formulierung konkreter Fragen bezüglich der Vorgehensweise während des Simulationsprozesses wider. Die Antworten entsprechen den zu dokumentierenden Artefakten und Vorgängen. Erkenntnisse und Feedback aus der Anwendung der Checkliste durch die Simulation Use Cases wurde in die Prozessbeschreibung und die Checkliste aufgenommen. Abbildung 25 zeigt exemplarisch einen Auszug aus der in GitLab hinterlegten Checkliste für die Simulation Use Cases. In der Abbildung ist dazu der erste Teil von Phase zwei des Credible Simulation Process mit seinen Teilschritten und den jeweils dazu gehörigen Fragen abgebildet.

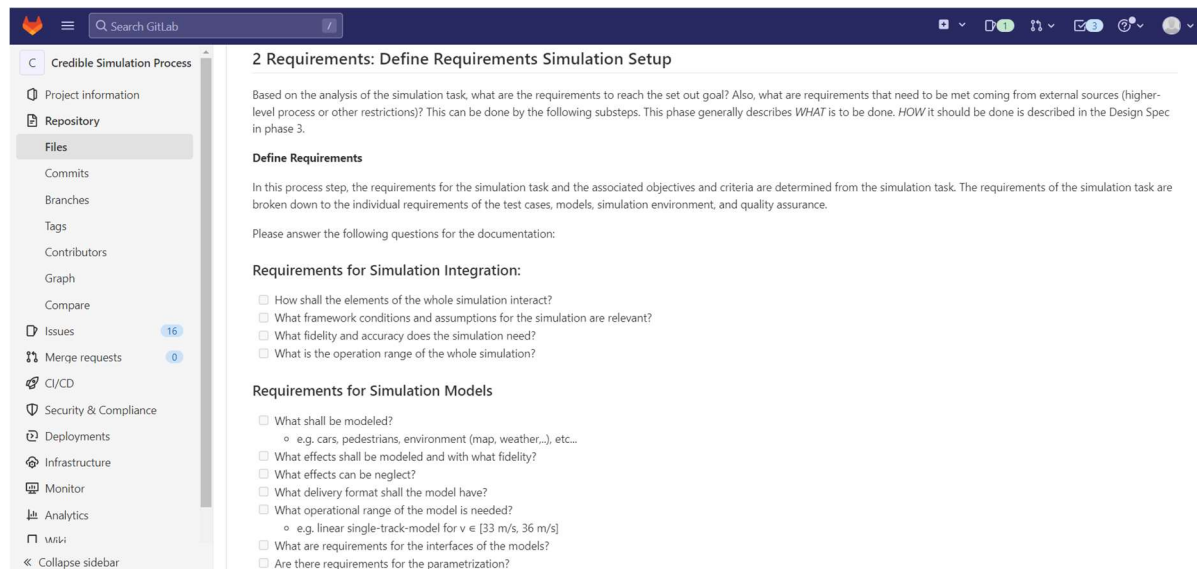


Abbildung 25: Auszug aus der Checkliste des Credible Simulation Process

Einer der Grundgedanken des Credible Simulation und Credible Modeling Process ist die Durchgängigkeit und Nachvollziehbarkeit einer Simulationsaufgabe prozessgestützt sicherzustellen. Um dies auch toolgestützt durchzuführen, wurde für die Anwendung des Prozesses im SET Level Projekt die Möglichkeit geschaffen, diesem im, in AP4.1 entwickelten Traceability Demonstrator, abzubilden.

In Zusammenarbeit mit dem AP4.1, wurde dazu das Erstellen von Projekten aus Credible Simulation und Credible Modeling Process Templates mit den Phasen und Schritten der Prozesse in TRACY eingeführt. Abbildung 26 zeigt die Anlagemöglichkeit der Templates, sowie ein instanziiertes Template für den Simulation Use Case 1. Die Instanziierung des Credible Simulation Process wurde im Rahmen des SET Level Projektes zur Unterstützung der Dokumentation der Simulation Use Cases im AP2.1 verwendet. Feedback aus dieser Anwendung ist in die finale Version des Credible Simulation Process mit eingeflossen.

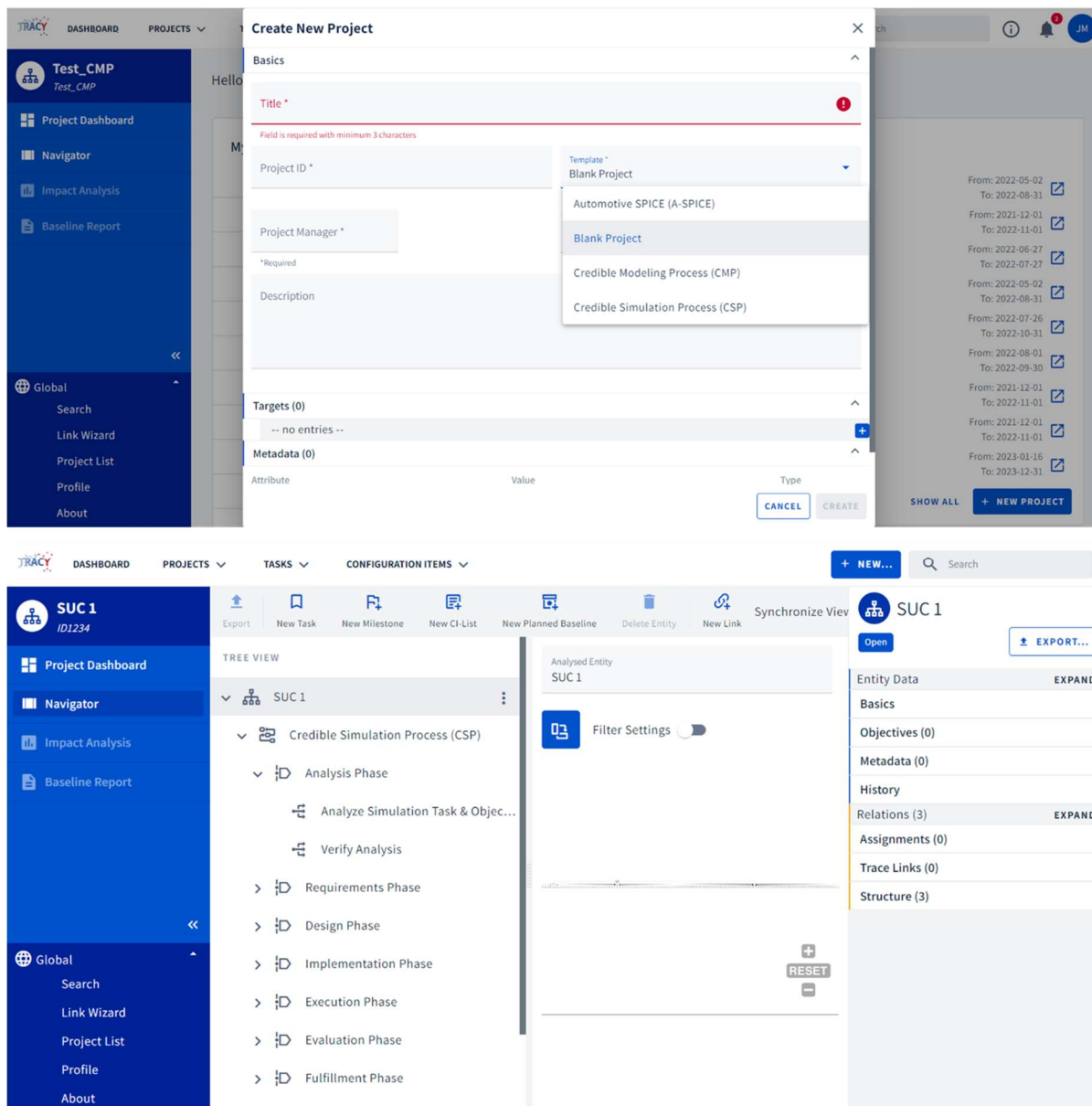


Abbildung 26: Anlagemöglichkeiten der Templates für Credible Simulation und Credible Modeling Process in TRACY (oben) und instanziiertes Template für den Credible Simulation Process in TRACY mit Subschritten (unten)

Zusätzlich zu den beiden Prozessen, welche die Durchgängigkeit und Nachvollziehbarkeit einer Simulationsaufgabe sicherstellen (Credible Simulation Process und Credible Modelling Process), wurde die „Preparation Phase“ des prostep ivip SmartSE Reference Process [1] erweitert und ebenfalls basierend auf einer Markdown-Ablage im Confluence gereviewt. Daraus wurde der „Agreement Process“ erstellt. Er ist als Ergänzung zu den beiden „Credible“-Prozessen zu sehen und geht diesen voran. Er dient zur Abstimmung rechtlicher, zeitlicher und inhaltlicher Rahmenbedingungen vor dem eigentlichen Projektstart. Auch für diesen Prozess wurde zur Vereinfachung der Anwendung wieder eine Checkliste erstellt und in GitLab abgelegt. Zusammen mit den beiden „Credible“-Prozessen bildet er auf Basis des SmartSE Referenzprozesses [1] den im SET Level Projekt definierten Kooperationsprozess. Dabei dient der SmartSE Referenzprozess [1] um den Koordinierungsvorgang zwischen den Partnern zu beschreiben. Abbildung 27 zeigt die Beziehungen und das Zusammenwirken der „Credible“-Prozesse mit dem prostep SmartSE Referenzprozess [1]. Zusätzlich ist hierbei auch die Top-Level-Sicht Agreementphase mit abgebildet.

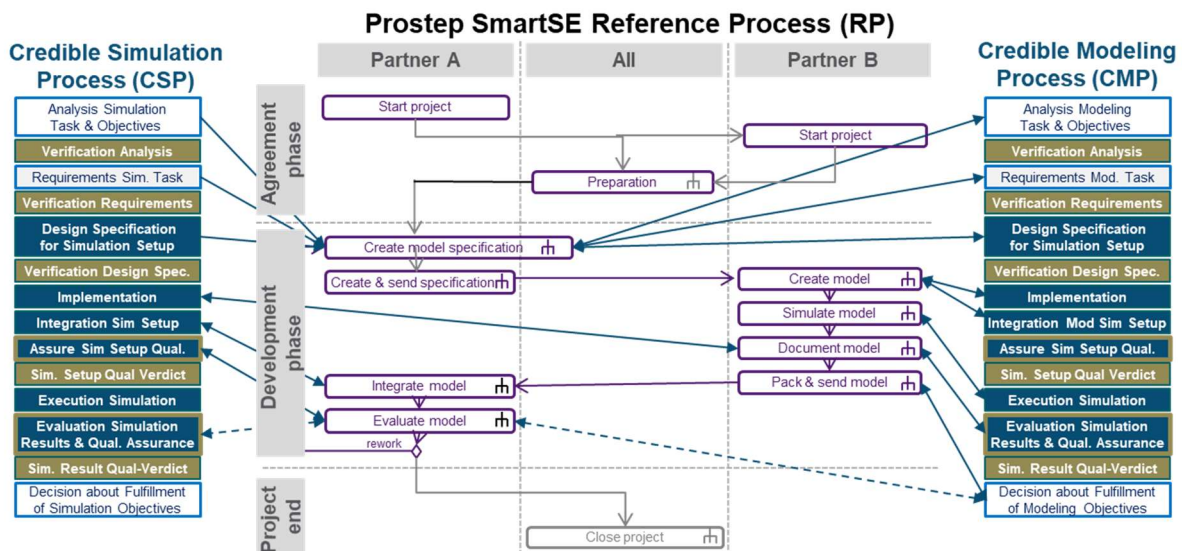


Abbildung 27: Zusammenwirken der „Credible“-Prozesse mit dem prostep SmartSE Referenzprozess [1] inklusive Top-Level-Sicht der „Agreementphase“

Der in dieser Form entworfene Kooperationsprozess wurde für eine Erprobung im Projekt zusammen mit dem AP5.2 vorbereitet. In Zusammenarbeit mit AP5.2 wurden Workshops zur Erprobung des Kooperationsprozesses vorbereitet und durchgeführt. Dazu wurden fünf Paaren bestehend aus je zwei Industriepartnern (OEM und Tier) gebildet. Mit diesem Paaren wurden jeweils separate Workshops durchgeführt und von PROSTEP begleitet. Die Zusammenarbeit und Dokumentation fanden online über MS-Teams und das Online-Kollaborations-Tool Miro statt. Die Ergebnisse der Workshops pro Industriepartnerkombination wurden im Nachgang ausgewertet und anonymisiert, um sie dann erst den Workshopteilnehmern zum Review und dann dem Projekt zur Verfügung stellen zu können. Das Feedback zu den Prozessen konnte über PROSTEP direkt zurückfließen, da sie an deren Erstellung beteiligt war. Ein Ausschnitt aus einer Übersicht im Online-Kollaborations-Tool Miro ist in Abbildung 28 zu sehen. Aus Vertraulichkeitsgründen sind die Inhalte der Abbildung weichgezeichnet worden.

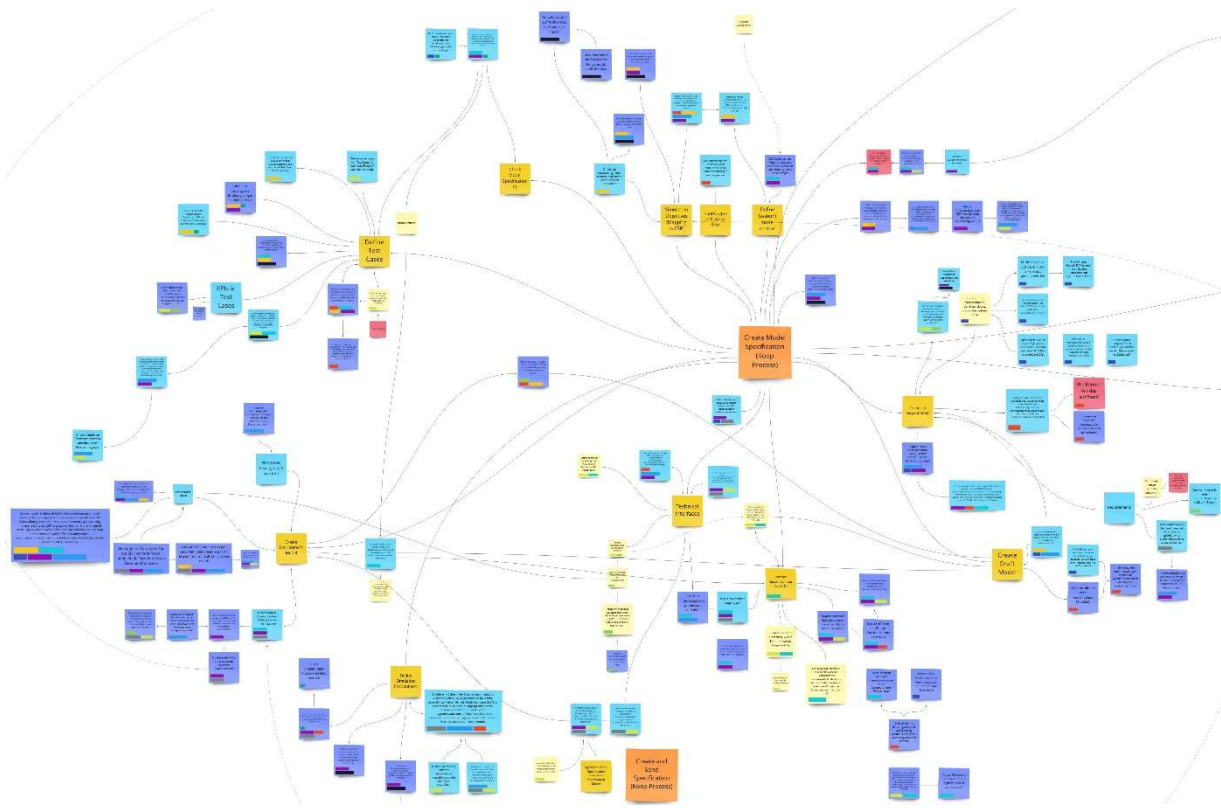


Abbildung 28: Ausschnitt aus der Konsolidierung der Ergebnisse zum Kooperationsprozess (Aus Vertraulichkeitsgründen weichgezeichnet)

Ähnlich zur Erprobung des Kooperationsprozesses wurde mit der Erstellung und Erprobung des Simulation Request (Simulationsauftrag) begonnen. Der Simulation Request ist der Ausgangspunkt für eine Absprache zwischen zwei Partnern, um eine Simulation zu beauftragen. Die im Simulation Request enthaltenen Informationen dienen als Input des Credible Simulation Process. Das zugehörige neutrale Austauschformat leitet sich aus dem bestehenden STMD (Simulation Task Meta Data) ab. Nach einigen für alle Partner offenen Absprachen zur Gestaltung des Simulation Request fand eine Erprobung in Zusammenarbeit mit AP5.2 statt. Dazu wurden Workshops zwischen den teilnehmenden Industriepartnern abgehalten, in denen Anforderungen an eine reale Umsetzung des Simulation Request im Zusammenspiel eines Auftraggebers und Auftragnehmers durchgespielt werden. Moderiert wurden die Workshops von PROSTEP als neutralem Partner. Die Ergebnisse der Erprobung wurden anonymisiert an das Projekt übergeben.

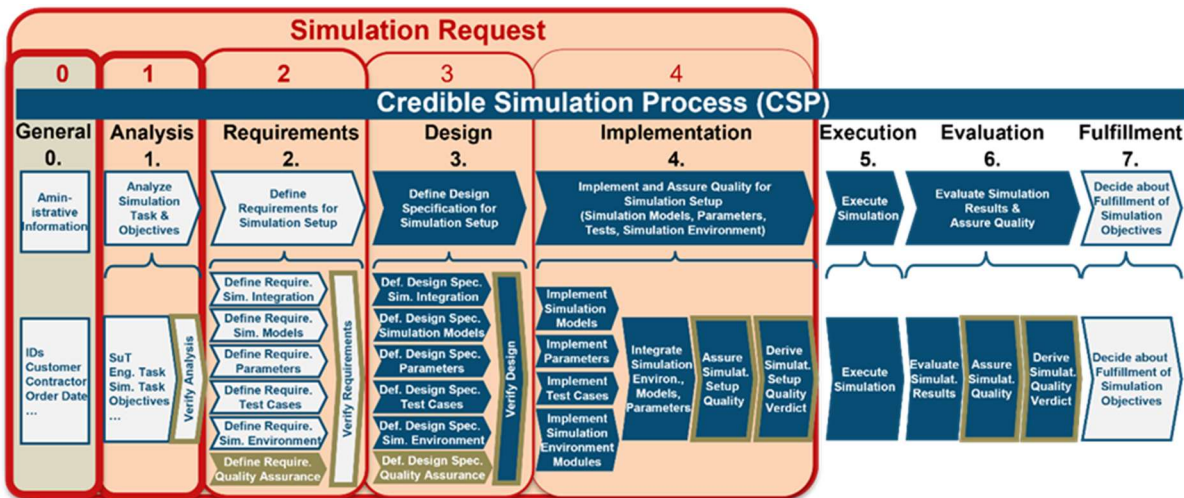


Abbildung 29: Simulation Request im Zusammenspiel mit dem Credible Simulation Prozess

1.1.3.2 AP 3.2 Modellauswahl und Integration

Die in AP3.2 geplanten Aufwände wurden im Projektverlauf, nach Absprache mit der TP-Leitung im AP3.1 geleistet. Jedoch wurde durch die Nutzung von SRMD (Simulation Resource Meta Data) für Modelle der Modellauswahlprozess unterstützt. (vgl. AP3.3 und SET Level Gesamtschlussbericht TP3)

1.1.3.3 AP 3.3 Datenmanagement

Der toolunabhängige SSP-Standard der Modelica Association ermöglicht den Austausch von aus FMUs bestehenden Systemen inklusive ihrer Parametrierung zwischen Simulationstools. Im SET Level Projekt wurde der Credible Simulation Process dazu zusätzlich so aufgebaut, dass er von seinen Phasen und Schritten zu der Erweiterung des SSP-Standards „STMD“ (Simulation Task Meta Data), einer Aktivität im Kontext „Glue Particle“ passend entwickelt wurde. Die Änderungen und Erweiterungen am Credible Simulation Process wurden dabei im engen Austausch mit den Aktivitäten zu den Glue Particles, auch außerhalb von SET Level (u.a. Modelica Association und prostep ivip SmartSE), vorgenommen. Besonders eng wurde dabei, bei der Entwicklung des STMD-Formats zusammengearbeitet, um das Mapping zwischen den Phasen und Schritten des Credible Simulation Process mit denjenigen des STMD XML-Schemas möglichst einfach und eindeutig zu gestalten. Die Dokumentation beider Entwicklungen wurde stetig abgeglichen, um Redundanzen zu vermeiden. Abbildung 30 zeigt einen Arbeitsstand aus dem Mapping zwischen Credible Simulation Process und dem XML Glue Particle Schema zum Datenmanagement der zusammen mit verschiedenen Partnern aus dem SET Level Projekt erarbeitet wurde.

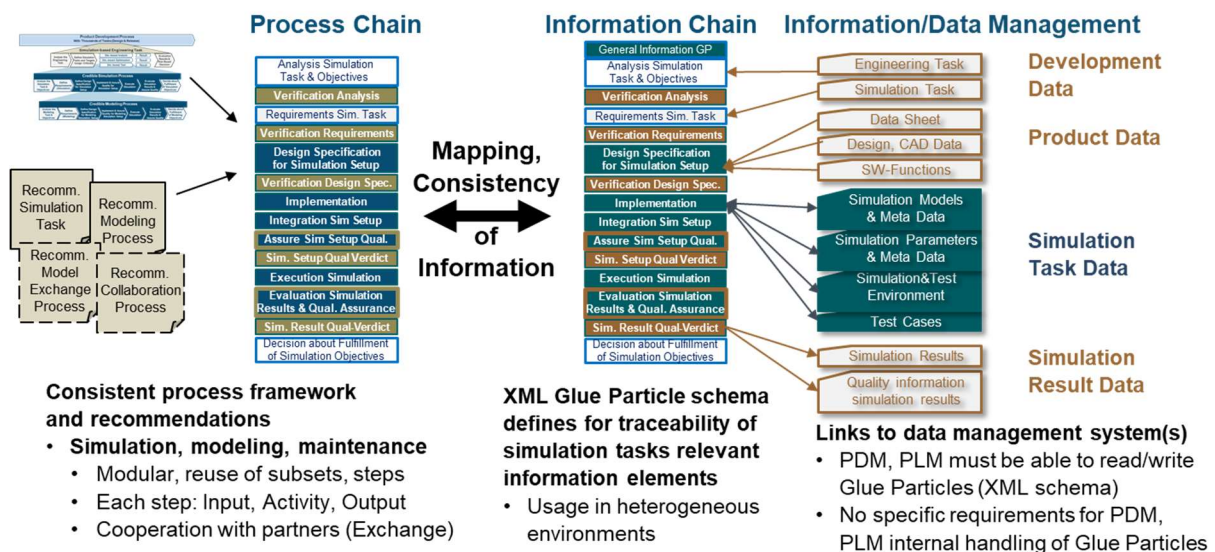


Abbildung 30: Konsistenz von Prozess- und Datenstruktur

Auch das Format SRMD (Simulation Resource Meta Data) stellt eine Erweiterung des SSP-Standards dar. Da auch dieses Format eng mit den Prozessen verknüpft ist, wurden Änderungen und Erweiterungen im engen Austausch mit den Aktivitäten zu den Glue Particles, auch außerhalb von SET Level (u.a. Modelica Association und prostep ivip SmartSE) unterstützt.

Im Rahmen des AP3.3 wurde darüber hinaus die Erstellung und Definition von Klassifikationen für die Beschreibung von konkreten Anwendungen des SRMD-Formates unterstützt. Dazu zählen die Beschreibung von konkreten Modellmetadaten, Metadaten über ein Tool und Metadaten für Karten & Szenarien.

Für die Beschreibung von konkreten Modellmetadaten wurde dabei an zwei Klassifikationen mitgearbeitet. In einer generischen Klassifikation werden die Metadaten zur Durchführung des Credible Modelling Process bereitgestellt. Dazu zählen die übergeordneten Informationen über das Modelloriginal und dessen Zielnutzung sowie Metadaten über die Maturität des Modells.

Zusätzlich sind auch spezifische Metadaten für die in SET Level erarbeitete Modelltypen: (z.B. Agentenmodelle, Sensormodelle, ...). Enthalten. Darüber hinaus wurde eine ISO spezifische Klassifikation definiert. Diese Klassifikation bietet die Möglichkeit die in der ISO-11010 definierte Nomenklatur zu Modelltypen als Metadaten zu nutzen. In Zuge dessen wurde auch die Metadatenversorgung von Modellen erprobt und zusammen mit dem AP2.1 validiert.

Die Beschreibung von Toolmetadaten soll die Nachverfolgbarkeit über die gesamte Simulationstoolkette unterstützen. Hierzu wurden die nötigen Metadaten identifiziert und insbesondere die im SET Level Projekt bereits vorhandenen gesammelt. Diese resultierende Klassifikation enthält dabei Metadaten über das Tool, wozu es entwickelt wurde und ISO 26262 toolbezogene Informationen.

Eine Beschreibung von Metadaten für Karten und Szenarien soll das Finden von Karten und Szenarien in einer Datenbank unterstützen. Als Vorarbeit dazu wurden verschiedene Quellen bereits bestehender Systeme untersucht (z.B. OpenStreetMap), in wie weit sich damit die Anforderungen von SET Level erfüllen lassen. Aufbauend auf den dadurch gewonnenen Erkenntnissen wurde die Klassifikation für Karten und Szenarien definiert. Diese Klassifikation enthält dabei Metadaten über typische Eigenschaften einer Szene, wie z.B. der Anzahl und Typen von Verkehrsteilnehmern aber auch der Art von Fahrmanövern.

Die genaue Definition der oben beschriebenen Klassifikationen und deren spezifizierten Metadaten sind unter [SETLevel SRMD and classifications for metadata.pdf · main · Public Projects / Credible Simulation Process Framework and Traceability Data / Traceability Data · GitLab](#) veröffentlicht worden. Zum Teil wurden die Inhalte zur weiteren Standardisierung an das prostep ivip Gremium SmartSE transferiert.

Durch die Konkretisierung des SRMD-Formates konnte zusätzlich die Grundlage für die Vorauswahl der Modelle im AP3.2 entwickelten Modellauswahlprozess geschaffen werden, da diese hauptsächlich auf dem für die Modelle hinterlegten Metadaten Attributen, wie sie z.B. im SRMD zur Verfügung gestellt werden basiert.

Für die Traceability und Anwendbarkeit der Prozesse ist es entscheidend dass die Informationen innerhalb der Prozesse (CSP, CMP, ...) durchgängig sind. Auch für die benötigten Datenartefakte wie Modelle, Szenarien, Simulationsauftrag sollte eine zum Prozess konsistente standardisierte Syntax und Semantik vorhanden sein. Dies wurde durch die im Projekt SET Level erarbeiteten Formate STMD und SRMD ermöglicht. Für die Konsistenz von Prozess- und Datenstruktur wurden zusammen mit TP4 Anforderungen an das Tooling von Metadaten erstellt. Da in TP3 kein separates Tooling erstellt werden sollte, wurden diese Inhalte als Teil des in AP4.1 entwickelten Traceability Demonstrators umgesetzt.

In enger Zusammenarbeit mit dem AP 4.1 wurde das Lesen und Schreiben des Formats STMD (Simulation Task Meta Data) im Traceability-Demonstrator TRACY dem neuesten Stand des Schemas angepasst und auch das Lesen und Schreiben des Formats SSP (System Structure and Parameterization) ermöglicht. Dies schafft eine standardisierte Austauschmöglichkeit für Simulationsaufgaben, die nach dem Credible Simulation Process entwickelt wurden und im Traceability Demonstrator dokumentiert sind.

Darüber hinaus wurde ebenfalls in enger Zusammenarbeit mit dem AP4.1 das Lesen und Schreiben des Formats SRMD (Simulation Resource Meta Data), in welchem Metadaten zu Modellen gefasst werden, umgesetzt. Dies ermöglicht dem Nutzer über TRACY in GitLab hinterlegte SRMD-Files zu bearbeiten. Die in Abbildung 31 dargestellte Benutzeroberfläche des Traceability Demonstrators TRACY zeigt einen Überblick über die Darstellung und Editierbarkeit von Metadaten am Beispiel „OsiPedestrian“.

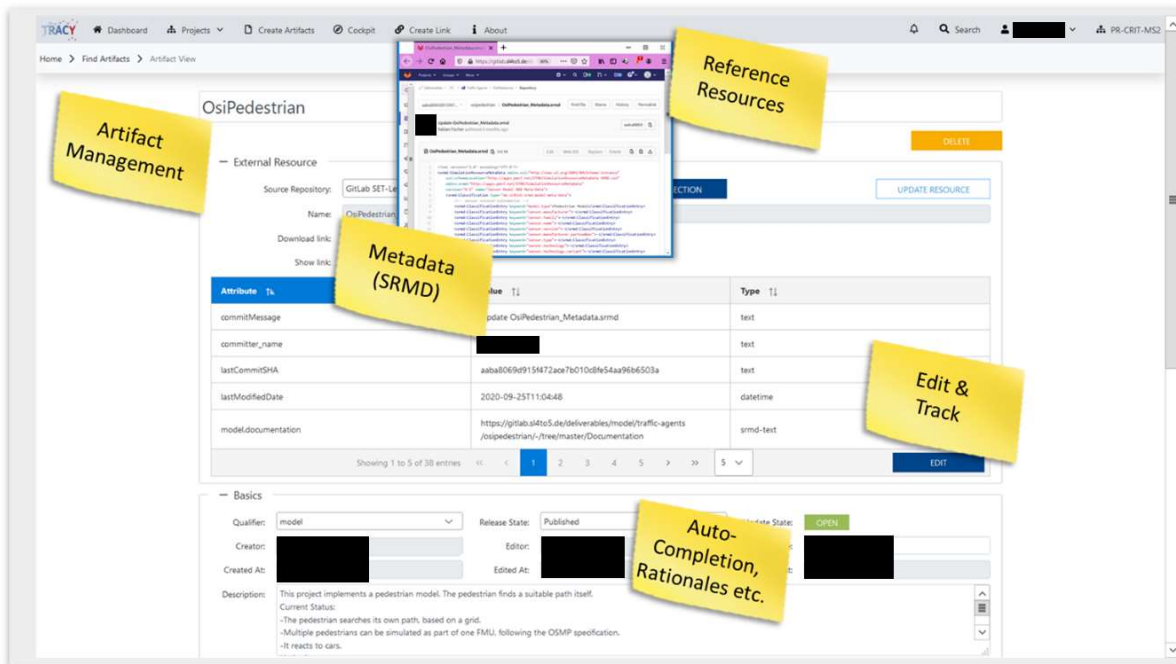


Abbildung 31: Übersicht über die Darstellung und Editierbarkeit von Metadaten im Traceability Demonstrator TRACY (Persönliche Daten geschwärzt)

Zusätzlich konnte durch die konkrete toolgestützte Anwendung der Formate im SET Level Projekt (z.B. Im- und Export oder direktes Editieren von Metadaten) eine Validierung der Einsatzmöglichkeiten speziell auch durch die damit verbundene Erprobung im industriellen Umfeld durchgeführt werden.

1.1.4 Teilprojekt 4: Instanziierung von Werkzeugketten für Entwicklung und Testen

1.1.4.1 AP 4.1: Ableitung der Schnittstellenanforderungen

Prozesse und Kollaboration

Prozesse und Workflows für die Kollaboration

Modelle von Sensoren, Aktoren und insbesondere der Fahrfunktion sind im Kontext von SET Level Verhaltensmodelle und als solche in Software implementiert. Auch wenn diese Softwaremodule als FMU umgesetzt werden, ihre Schnittstellen dem OSI-Standard folgen und der Zusammenbau mittels SSP definiert wird, gibt es doch immer wieder den Bedarf, diese Schnittstellen zu ändern. Formale Änderungen der Syntax (wieviele Parameter welchen Typs hat ein bestimmter Aufruf?) sind dabei noch harmlos in dem Sinn, dass sich durch geeignete Werkzeuge überprüfen lässt, ob die Änderung an wirklich allen relevanten Stellen berücksichtigt wurde. Viel diffiziler sind Änderungen an der Semantik, also der Bedeutung von Parametern oder Methoden im Kontext der Simulation.

So wurde bereits sehr früh im Projektverlauf deutlich, dass ein definierter und gesteuerter Prozess zur Änderung von Schnittstellen für die erfolgreiche Zusammenarbeit im Projekt unverzichtbar ist.

Deshalb entwickelten AP 2.1 und AP 4.1 gemeinsam einen *Schnittstellenänderungsprozess*, der seitens der Infrastruktur ein Issue-Management-System (hier: GitLab) und ein Code-Versionierungssystem (hier: Git) nutzt. In betreffenden Repositories wurden Change Control Boards (CCBs) etabliert, um Priorisierung, Implementations- und Integrationsplanung der Änderungen abzustimmen. Insbesondere die extern wirksamen Beiträge zur Weiterentwicklung von OSI/OSMP erforderten dabei ein besonders sorgfältig abgestimmtes Vorgehen.

Der Schnittstellen-Änderungsprozess zur nachhaltigen Sicherung der Passung zwischen Modellkomponenten wurde dabei anhand eines Beispiels gemeinsam entwickelt und wie folgt definiert:

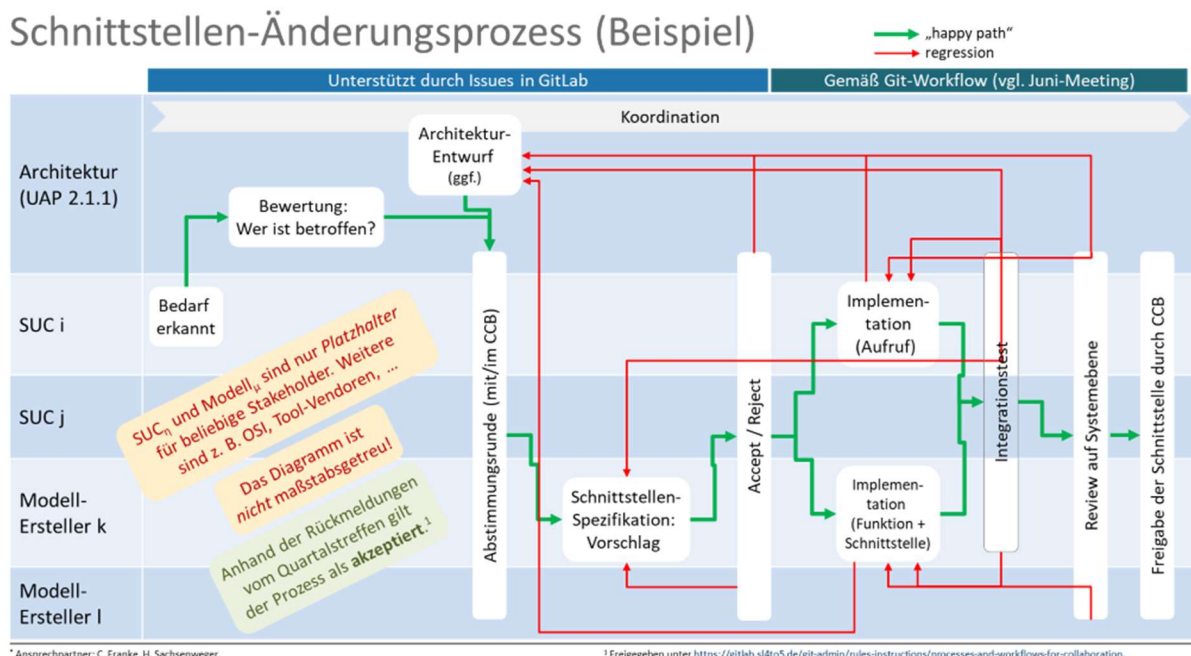


Abbildung 32: Schnittstellen-Änderungsprozess (dargestellte Phasengrößen entsprechen nicht dem real nötigen Zeitaufwand)

Dieser wurde beim Quartalstreffen im September 2020 vorgestellt, diskutiert und in Kraft gesetzt.

Spezifikation und Ramp-Up der projektweiten GitLab-Installation

Ursprünglich war in der Vorhabensbeschreibung (VHB) davon ausgegangen worden, dass eine Datenbank für Szenarien benötigt wird. Sehr schnell reifte jedoch die Erkenntnis, dass eine leistungsfähige Versionsverwaltung für *jedwede Art* von Artefakten benötigt wird, die im Projekt entstehen. Selbst das Handling interner Dokumente zum Projektmanagement würde davon profitieren. Bei vielen Projektpartnern lagen bereits Erfahrungen mit der freien Software „Git“ vor.

Im Rahmen von AP 4.1 wurde daher ein Prozess moderiert, der über eine Anforderungsermittlung und eine Bewertung diverser technischer Optionen zur Entscheidung führte, Git in der Inkarnation „GitLab“ zu verwenden. Neben der reinen Versionsverwaltung standen dem Projekt damit auch mächtige Funktionalitäten zur Verwaltung hierarchisch gruppierter Repositories, zur Koordination der Zusammenarbeit via Issue-Tracking und für individuelle Benutzerberechtigungen zur Verfügung.

Für den GitLab-Einsatz wurde je ein Handbuch für Standardnutzer und für Repository-Verantwortliche mit spezifischem Regelwerk erstellt. Zusätzlich entstanden Anleitungen für Anwender mit Beispielen, wie mit bestimmten Artefakten umgegangen werden soll.

Im weiteren Verlauf des Projekts wurden die Projektteilnehmer kontinuierlich bei der Verwendung von GitLab unterstützt. Es wurden „Lessons Learned“ zum GitLab durchgeführt, aus denen deutlich erweiterte Versionen der Handbücher hervorgingen.

Manual zum Aufbau von und Umgang mit Repositories für Traffic Spaces und Scenarios

Im Rahmen von SET Level musste zunächst geklärt werden, wo genau Referenzszenarien gespeichert werden, ob das Mehrebenen-Konzept als Datenbanklösung nur die tiefste Ebene konkreter Szenarien oder auch höhere Ebenen funktionaler Szenarien mit Parameterräumen oder verbale Beschreibungen von Szenarien oder gar die Rohdaten aus konkreten Versuchsfahrten erfordert, ob diesbezüglich auf die PEGASUS-Datenbank aufgesetzt werden kann, ob ein eigenes Repository geschaffen werden muss oder ob durch Referenzierung auf Szenarien-Repositories der IT-Vendoren in ausreichendem Maße Beispiele durchgespielt werden können, die vor dem Hintergrund der Brauchbarkeit der Schnittstellenformate, von Austauschbarkeit und Wiederverwendung den Anforderungen des Testkonzeptes genügen.

Nach einer Analyse der relevanten Anforderungen wurde beschlossen, eine eigene Sammlung von Traffic Spaces und Scenarios in GitLab aufzubauen.

Gemeinsam mit TP 1 hat AP 4.1 daher ein Handbuch entwickelt, das beschreibt, wie Traffic Spaces und die darauf aufbauenden Scenarios in Gruppen und Repositories des GitLab abzuliegen sind (sog. „Ablagestruktur“), welches Benennungsschema anzuwenden ist und wie die unterschiedlichen Bedürfnisse der drei SUCs befriedigt werden können.

Die folgenden Abbildungen sind Beispiele dieser Ablagestruktur:

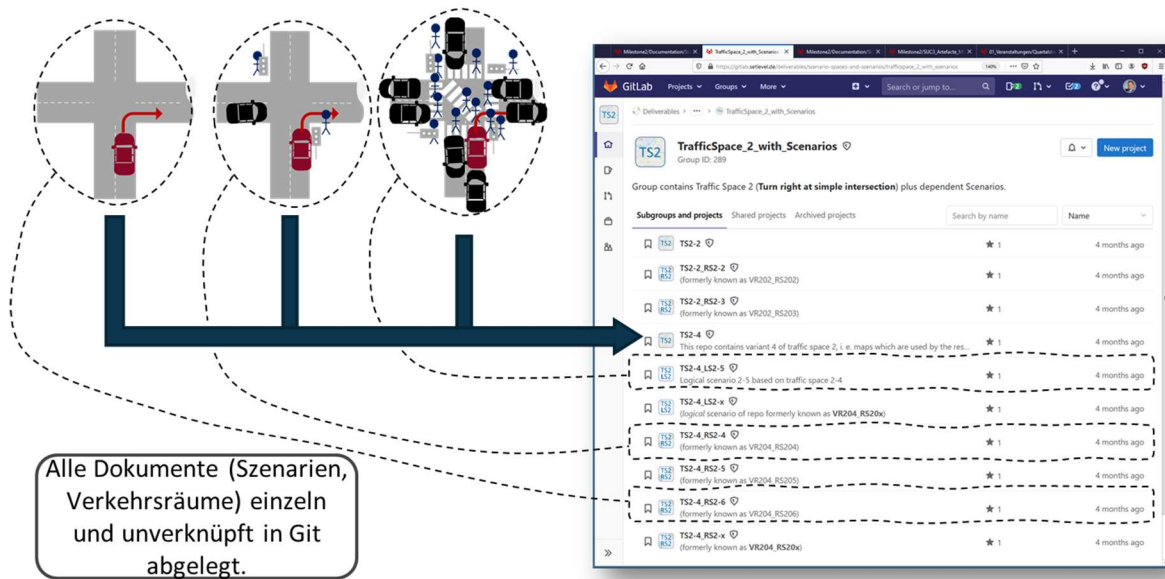


Abbildung 33: Ablage der Traffic Spaces in GitLab

Auf diese Traffic Spaces wird in den CSP-konformen Dokumentationen der SUCs, die mit Unterstützung von AP 2.1 erstellt wurden, per URL verwiesen (Abbildung 34):

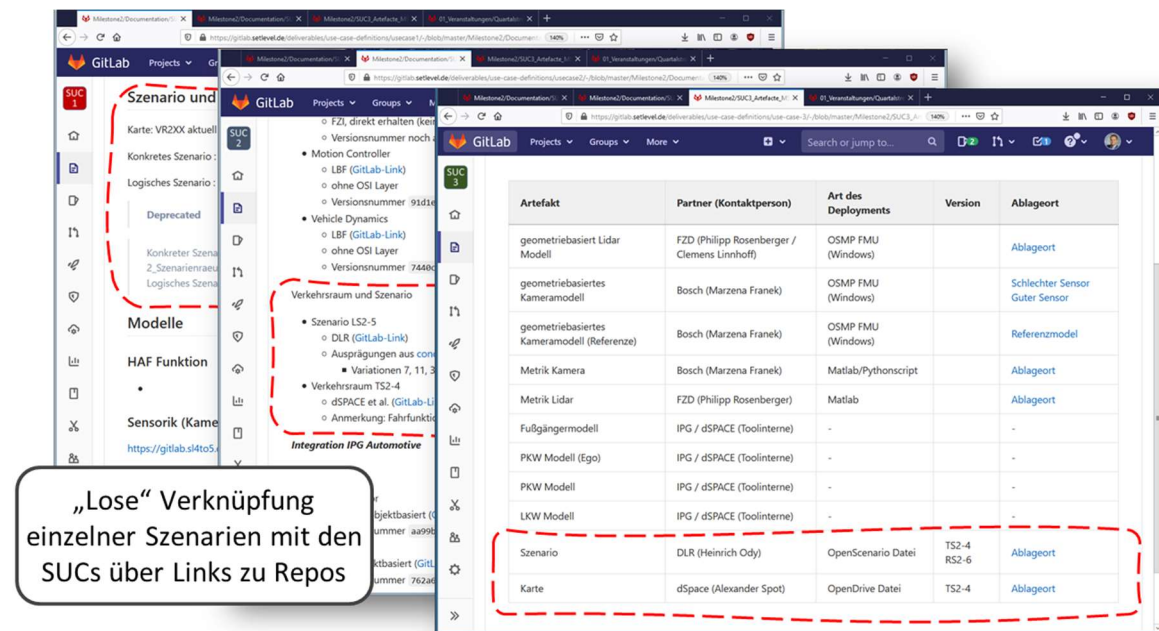


Abbildung 34: SUC und Szenarien verknüpft

Somit ist ein Rückwärtsverweis vom SUC auf seine Quellen gegeben.

Betrachtet man jedoch umgekehrt eines der Traffic-Space-Repositories, so ist *nicht* ersichtlich, in welchen Simulationen dieser Traffic Space verwendet wird:

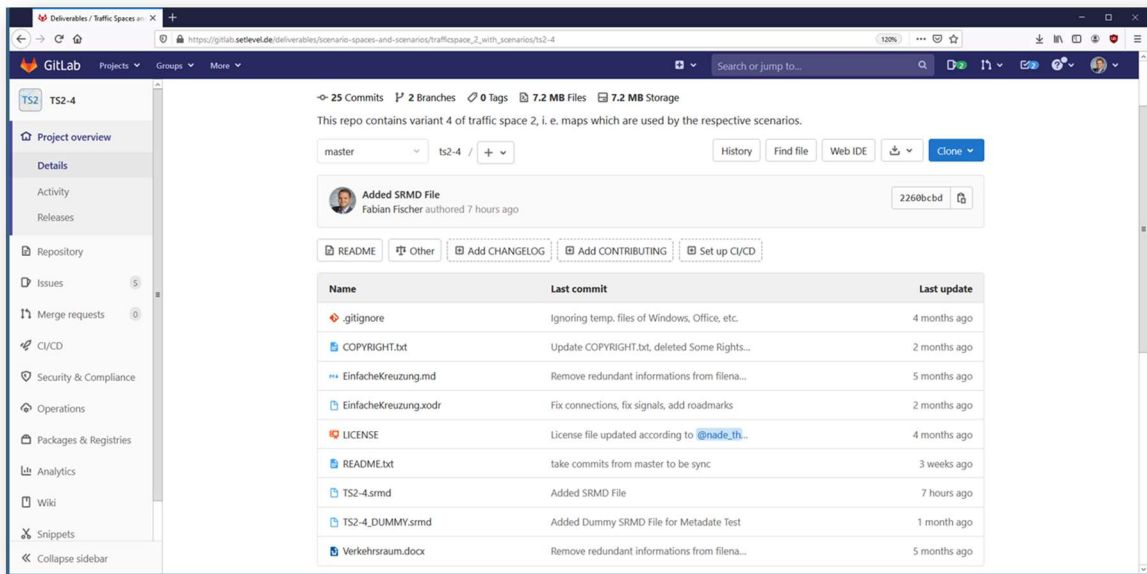
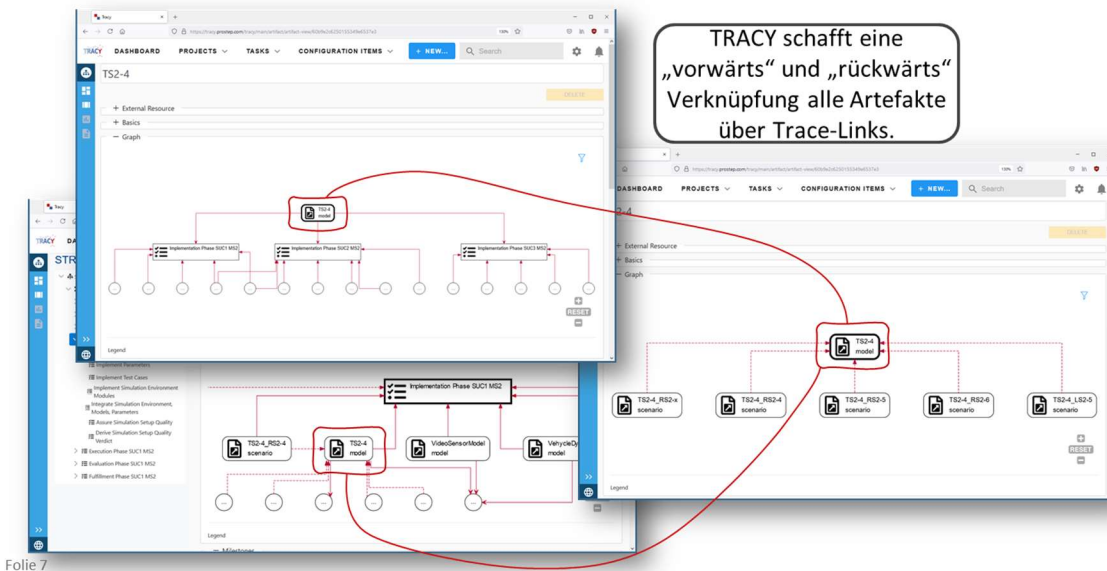


Abbildung 35: Traffic Space Repository

Benötigt wird hier eine *Vorwärtssuche*.

Die Traceability-Software „TRACY“ ermöglicht es in diesem Fall, über die Verknüpfung von SUC und Traffic Space via Trace-Links sowohl eine Vorwärts- als auch eine Rückwärtssuche:



Folie 7

Abbildung 36: Trace-Links in TRACY

Mittels der dokumentierten Ablagestruktur und der Software TRACY konnte eine effiziente projektweite Zusammenarbeit zwischen den SUCs (TP 4) und den APs, die sich mit Verkehrsräumen, Referenzszenarien und erweiterten Szenarienbeschreibungssprachen (jew. AP 1.2) und der Spezifikation der Szenarienbeschreibung (AP 2.1) befassen, etabliert werden.

Ab 2021 fanden Austauschgespräche statt mit ADScene sowie der University of Warwick, welche „Safety Pool“ bereitstellt. In beiden Projekten wird an Szenario-Datenbanken gearbeitet.

Eigenes und fremdes Geistiges Eigentum, Lizenzen

Die Entwicklung hoch automatisierter Fahrfunktionen erfordert eine sehr intensive Zusammenarbeit zwischen vielen Beteiligten, von Automotive OEMs über TIER1s und Dienstleistern hin bis zu Forschungsinstituten. Dabei müssen Rechte an Geistigem Eigentum (Intellectual Property, IP) allerdings immer gewahrt bleiben.

Dies gilt selbstverständlich auch für Rechte an Werken, die im Projekt SET Level verwendet werden, als auch für Werke, die im Rahmen von SET Level entstanden. Bei diesen Werken handelt es sich vor allem um folgende Typen:

- Software (im eigentlichen Sinne oder als Bestandteil von Verhaltensmodellen)
- Modelle von Verkehrsräumen und Szenarien
- Veröffentlichungen (wissenschaftliche Publikationen, Vorträge u.s.w.)
- Sonstige Dokumentationen von Projektergebnissen

PROSTEP und BMW förderten daher gemeinsam eine entsprechende Diskussion und Bewusstseinsbildung bei den Projektteilnehmern. BMW benannte Experten, die als Ansprechpartner zur Verfügung standen. Es zeigte sich, dass für die weitere Nutzung von Projektergebnissen im Allgemeinen die aus der Software-Welt stammende Lizenzbildung „Creative Commons Attribution-ShareAlike 4.0 (CC BY-SA 4.0)“ angemessen ist. Sie wurde projektweit empfohlen, wobei jeder Urheber unvermindert das Recht behielt, sich für andere Regelungen zu entscheiden. Es wurde allerdings verlangt, dass diese Entscheidung zusammen mit dem Werk, also im selben Repository, dokumentiert wird. PROSTEP prüfte die Einhaltung dieser Regel regelmäßig über einen wesentlichen Zeitraum des Projekts hinweg.

BMW engagierte sich für die Dokumentation der zu Modellen, Szenarien und Karten notwendigen Lizenzen. Beispielhaft wurden daher ALKS-Szenarien auf der TPX-Website bereitgestellt und hierfür die o. g. Lizenzbildung CC-BY-SA-4.0 ausgewählt.

Architektur

Im Bereich „Architektur“ arbeitete das AP 4.1 eng mit AP 2.1 sowie mit den APs 4.2 bis 4.4 (den drei SUCs) und TP 3 (numerische Simulationsmodelle) zusammen.

Eine der Grundmotivationen des Projektes SET Level rührte aus der Erkenntnis, dass fortgeschrittene Fahrassistenzsysteme (ADAS) längst nicht mehr ausschließlich durch physikalische Tests verifiziert werden können. Simulation ist in sehr großem Umfang erforderlich. Eine integrative Simulationsarchitektur von Sensormodellen über Verkehrsszenarien bis hin zu Werkzeugen und Prozessen bietet eine Grundlage, um die entstehenden technischen und organisatorischen Herausforderungen bei der Simulation zu meistern. Sie hilft, Synergien zu nutzen und redundante Arbeiten sowie Fehlerquellen zu vermeiden, insbesondere in großen, heterogenen oder verteilten Teams.

Solch eine Simulationsarchitektur muss also insbesondere

- Orientierung bieten,
- eine hierarchisch beliebig tiefe Dekomposition je nach Simulationsziel ermöglichen,
- Bereiche der Architektur zulassen, die aus IP-Gründen nicht offengelegt werden,
- entsprechend modular aufgebaut sein,
- Definition und Standardisierung von Schnittstellen unterstützen,
- je nach Simulationsziel und Kontext unterschiedliche Sichten verschiedener Detaillierungsstufen ermöglichen.

Welche *Schnittstellen* sind hier gemeint?

- viele ganz unterschiedliche:
 - fahrzeugintern zwischen *Komponenten*,
 - Fahrzeug zur *Verkehrsumgebung*,
 - Fahrzeug und Umgebung zum *Simulator*,
 - dieses *Simulationssystem* zur *Simulationsaufgabe*,
 - *Simulationssystem* und -aufgabe zum *Simulationsprozess*.
- ⇒ **Simulation an sich** wird als **ein System** betrachtet!

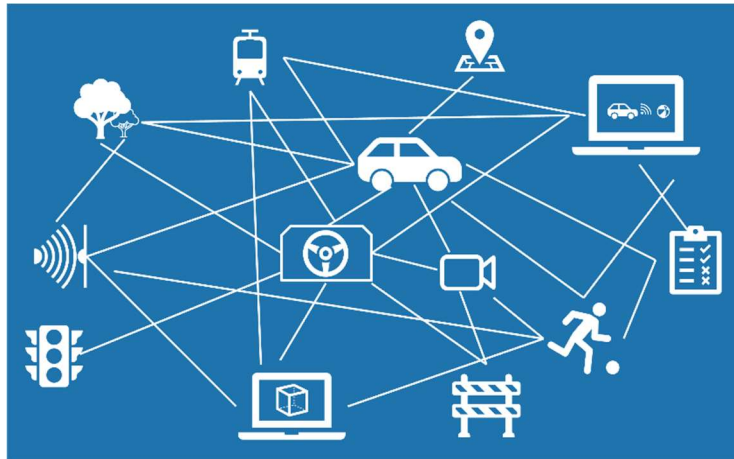


Abbildung 37: Welche Schnittstellen?

Allein im Bereich der Werkzeuge existiert eine Reihe relevanter Software-Systeme. Die folgende Abbildung führt nur einige davon auf:

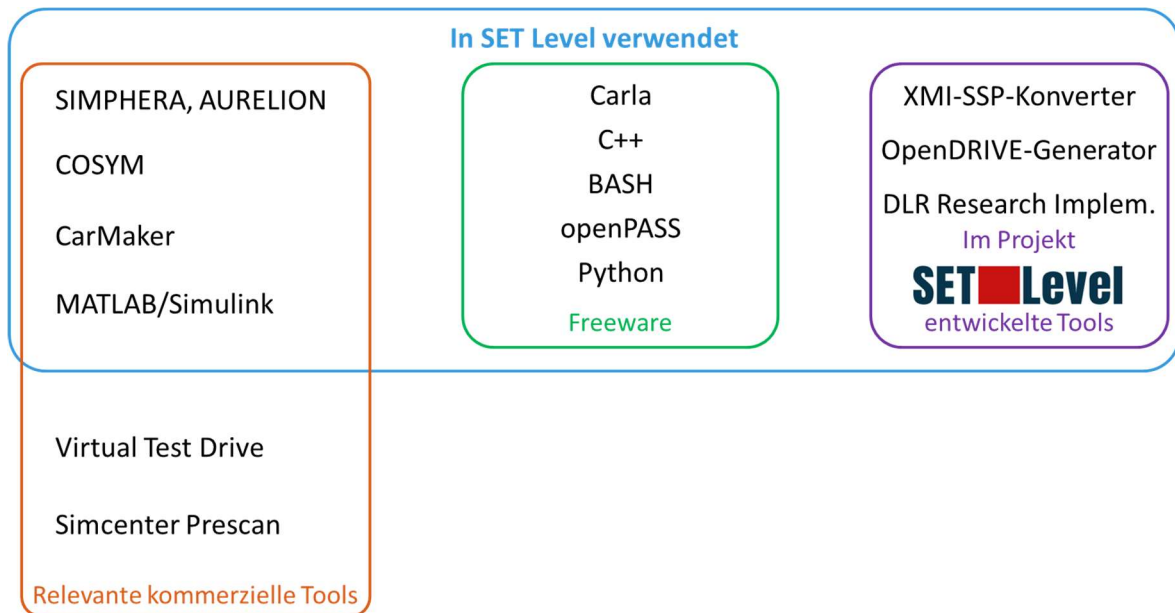


Abbildung 38: Beispiele für Software im Bereich ADAS-Simulation

In anderen Bereichen der Architektur ist die Vielfalt mindestens ebenso groß.

Im Projekt gingen wir die daraus erwachsenden Herausforderungen mit modernen Methoden an, die eine simulationsbasierte Analyse, Entscheidungsfindung (vgl. [3]) und Verifizierung ermöglichen. Die Methoden entstammen dem Model Based Systems Engineering (MBSE) und führen uns nahtlos von der abstrakten Systemarchitektur zu implementierbaren Simulationsmodellprototypen.

Um dieses Ziel zu erreichen, wenden wir das bewährte RFLP-Muster [4] auf das Gesamtsystem der Simulation als solcher an. Das Gesamtsystem besteht nicht nur aus dem Simulationsmodell, sondern auch aus dem Simulator und der gemeinsamen Einbettung in einen glaubwürdigen Simulationsprozess zur Erfüllung einer Simulationsaufgabe. Der Vielfalt und Komplexität der realen Verkehrsteilnehmer wird durch ein Rahmenmodell mit konfigurierbaren Komponenten Rechnung getragen.

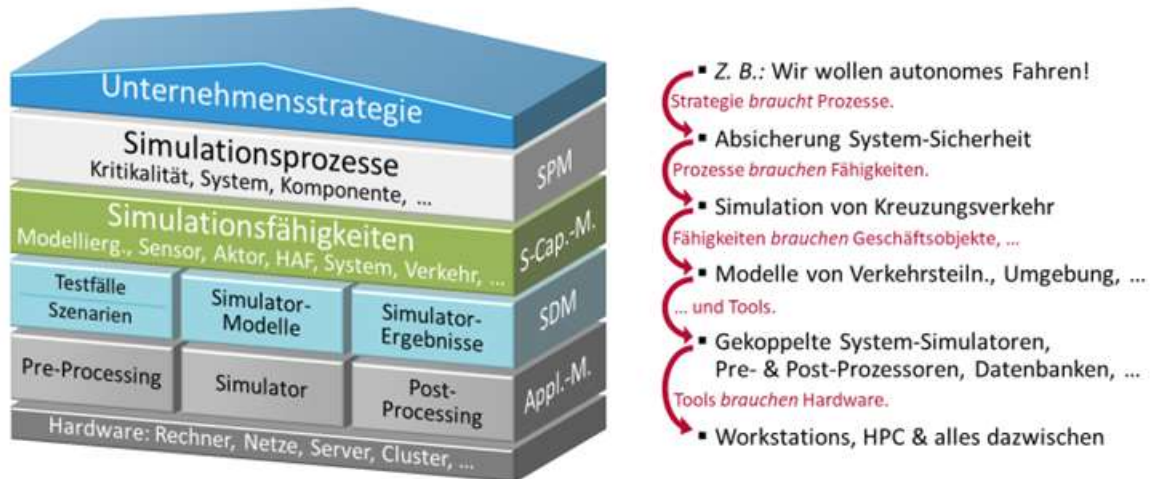
Die Architektur der untersten Abstraktionsebene kann direkt für den Zusammenbau des Modells in einer Simulationssoftware verwendet werden.

Als Modellierungssprache wird eine Teilmenge der standardisierten und weit verbreiteten Unified Modeling Language (UML) [5] verwendet. Dies garantiert, dass die Methode einfach zu verstehen und unabhängig vom Modellierungswerkzeug ist.

Belastbare Simulationsergebnisse erfordern eine Architektur, die eng mit einem glaubwürdigen Prozess verbunden ist, wie dem Credible Simulation Process (CSP).

Zusammen ergibt sich daraus auch ein konsistenter Beitrag zum Enterprise Architecture Modeling (EAM) der Organisation.

Unternehmensarchitektur umfasst mehrere Ebenen:



Folie 54

Vgl. Vortrag "EAM for CAE – Enabler für die Digitalisierung", Dr. Carsten Franke & Dr. Lars Wagner, SRAVEC Baden-Baden, 21.11.2018

Dr. Carsten Franke et al., 06.12.2019 – Klassifikation C2: Projekt intern

Simulationsbasiertes Entwickeln und Testen von Level 4 und 5 Systemen

Abbildung 39: Schematische Darstellung der Ebenen der Unternehmensarchitektur im Kontext Simulation

Eine klug gestaltete Unternehmensarchitektur bringt auch wirtschaftlich Vorteile mit sich: Je mehr Komponenten der Architektur allgemeiner verwendbar sind, umso besser lassen sich Entwicklungsgegenstände wiederverwenden, umso ökonomischer lassen sich Software-Lizenzen und Fortbildungsmaßnahmen einsetzen, umso weniger Overhead entsteht durch zusätzliche Schnittstellen, und umso kostengünstiger fällt eine Zertifizierung einer Tool-Architektur für Produktfreigaben aus.

PROSTEP hat in diesem Kontext sein Know-how im Bereich Unternehmensarchitektur und Systems Engineering eingebracht.

Dokumentation und Spezifikation der Integrationsarchitektur im AP 4.1 erfordern eine systematische und strukturierte Herangehensweise. Deshalb wurden Methoden der Modellierung von Unternehmensarchitektur herangezogen (siehe Abbildung 39).

Randbedingungen und Zielsetzung des Projekts bestimmen, dass von diesen Ebenen vor allem Prozesse, Daten und Werkzeuge, relevant sind. Es wurde identifiziert, dass sich diese drei Ebenen, wie abgebildet, auf die TPs 2, 3 und 4 des Projektes verorten (siehe Abbildung 40).

Architektur-Dreibein

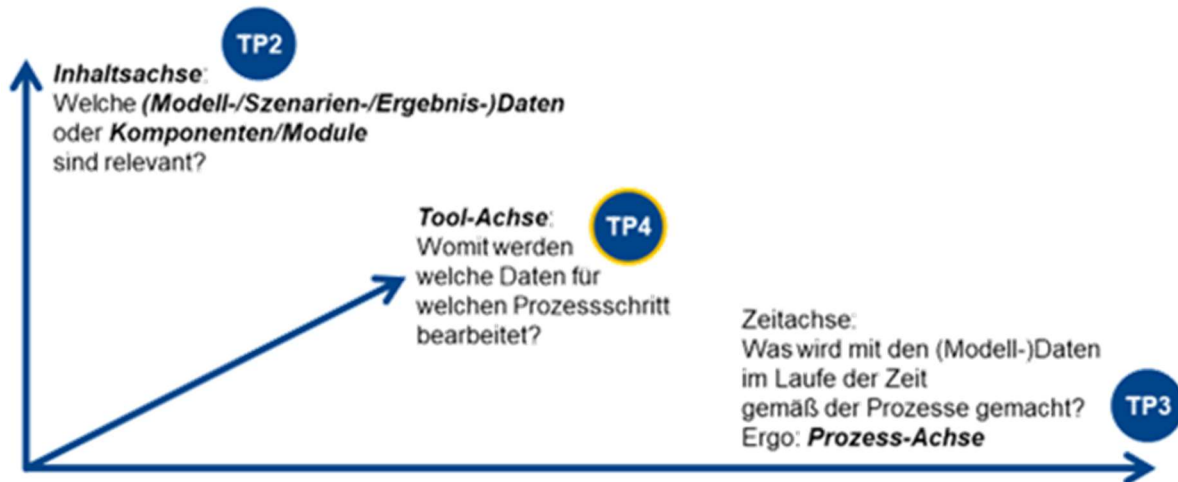


Abbildung 40: Architektur-Dreibein

Das AP 4.1 fokussierte sich demnach auf die Tool-Achse, benötigte für seine Arbeit aber auch sehr die Architektur der Inhalte. Deshalb wurde eine enge und intensive Zusammenarbeit mit AP 2.1 etabliert.

Methode der integrativen Architektur der Simulation

Dem allgemeinen Ansatz des Projekts entsprechend konnte es *nicht* darum gehen, eine feste, für alle Partner verbindliche Architektur der Simulation zu formulieren. Vielmehr musste eine *Methode* erarbeitet werden, die es jedem Partner ermöglicht, diese Architektur gemäß seinen Bedürfnissen auszugestalten und dabei von den beschriebenen Vorteilen zu profitieren.

An der Entwicklung der gesuchten *Methode der integrativen Architektur der Simulation* haben sich u. a. BMW, Conti ADC, DLR, OFFIS, Opel, PROSTEP, VW und ZF über mehrere Quartale hinweg beteiligt. Die intensive Zusammenarbeit der APs 2.1 und 4.1 bis 4.3 zeigte bald essenzielle Fragen auf, die nur gemeinsam gelöst werden konnten:

1. Wie ist sichergestellt, dass Architektur-Sichten (Diagramme) nachhaltig zusammenpassen? (vgl. Abbildung 41)
2. Welche Architektur-Sichten sind relevant für wen?
3. Wie lassen sich Prozess- und Modell-Architektur optimal miteinander in Verbindung setzen? (Bezug zum CSP, vgl. TP 3)

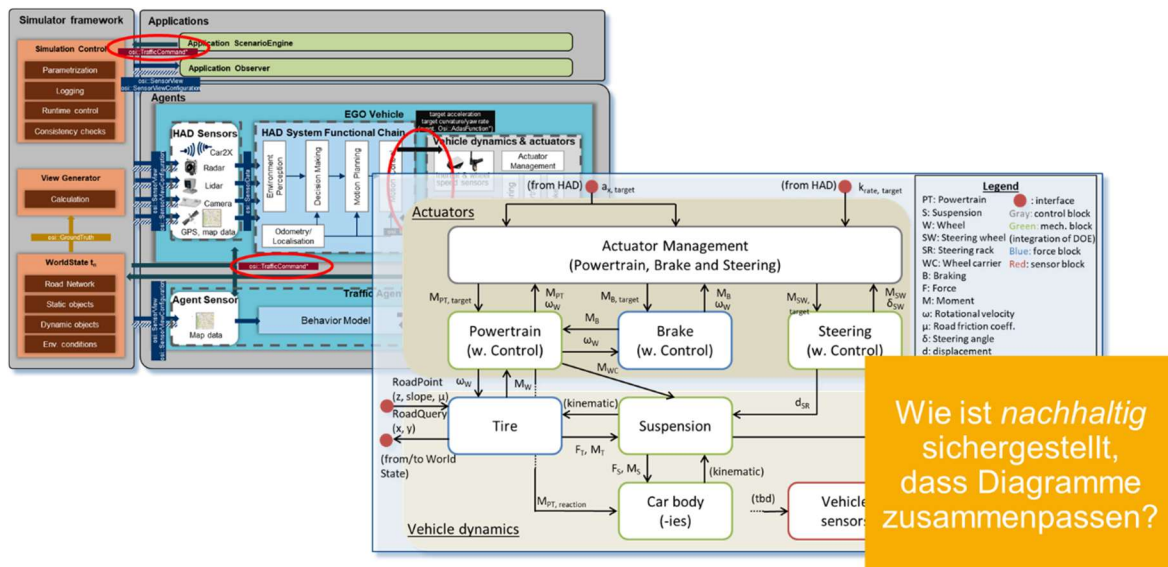


Abbildung 41: zur Frage 1 – Nachhaltige Passung von Architektur-Sichten

Die Passung der Architektursichten wurde durch den Einsatz eines einheitlichen Design-Werkzeugs (Sparx Systems' Enterprise Architect, kurz EA) zur UML-Modellierung erheblich unterstützt. Dafür gab es viele gute Gründe:

- Die schiere Größe des Modells zwingt zum Einsatz eines geeigneten Werkzeugs.
- Es überträgt automatisch Änderungen von einem Element / Diagramm auf alle anderen Vorkommen,
- sichert dadurch, dass die Konsistenz gewahrt bleibt,
- und hilft, mühsame Arbeit und Fehlerquellen zu reduzieren.
- Das UML-Modell kann die Architektur einer Simulation präzise und unmissverständlich beschreiben. Dadurch fördert es zugleich Bestrebungen zur Standardisierung.
- Ein grafisches „Inhaltsverzeichnis“ (vgl. Abbildung 46) verschafft einen guten Überblick.

PROSTEP hat ein dazu passendes Konzept zum Betrieb eines gemeinsamen Datenbankservers für den EA als *Single Source of Truth* entwickelt. Der Server wurde vom DLR betrieben.

Die Repräsentation in genau diesem Tool ist dabei nur als Beispiel zu verstehen. Andere SysML- und UML-Werkzeuge lassen sich hier genauso verwenden, denn es wurde im Sinne möglichst allgemeiner Verwendbarkeit darauf geachtet, möglichst grundlegende, gemeinsame und allgemein etablierte Konstrukte aus diesen Sprachen zu verwenden.

Neben der reinen Kollaboration förderte diese technische Infrastruktur auch die nachhaltige Konsistenzsicherung der Architektur-Sichten.

Zur *Methode der integrativen Architektur der Simulation* erschienen Präsentationen mit Artikeln beim NAFEMS World Congress 2021 und der VDI-Konferenz SIMVEC 2022.

Beim Abschluss-Event des Projektes war dem Thema der Marktstand B04a gewidmet. Entsprechend wurde ein Poster, ein Foliensatz und eine Vortragsaufzeichnung veröffentlicht.

Die Fortschritte und die Bedeutung des Themas „Architektur“ manifestierten sich darin, dass dazu in einer Serie von Quartalstreffen jeweils ein Hauptvortrag stattfand.

Model-based Systems Engineering (MBSE)

Mit zunehmender Komplexität eines Systems wird es immer schwieriger, alle Abhängigkeiten zwischen seinen Komponenten zu erfassen. Wie Flugzeuge und Satelliten bestehen auch hochautomatisierte Fahrzeuge aus einer Vielzahl unterschiedlicher Komponenten, die von Ingenieuren aus den Bereichen Mechanik, Elektrik/Elektronik und Software entwickelt werden. Systems Engineering (SE) ist ein interdisziplinärer Ansatz, um die Realisierung solcher

komplexen Systeme zu bewältigen. Er „integriert alle Disziplinen und Fachgruppen in eine Teamarbeit und bildet einen strukturierten Entwicklungsprozess, der vom Konzept über die Produktion bis zum Betrieb reicht“ [6]. Dies geschieht durch die Zerlegung des Systems in überschaubare Teilsysteme mit definierten Grenzen und Schnittstellen, die dann von den jeweiligen Spezialisten entwickelt werden können.

Eine der großen Herausforderungen ist nach wie vor die Koordination zwischen den verschiedenen beteiligten Disziplinen. Hierfür schlägt das INCOSE [7] den MBSE-Ansatz vor, der „die formalisierte Anwendung der Modellierung zur Unterstützung der Systemanforderungen, des Entwurfs, der Analyse, der Aktivitäten zu Verifikation und Validierung, beginnend in der Konzeptphase und fortlaufend über Entwicklung und spätere Lebenszyklusphasen“ darstellt [8]. Die bevorzugten Modellierungssprachen sind grafische Sprachen wie UML oder SysML [9], die den Vorteil einer universellen Darstellung des Systems über alle Domänen hinweg bieten.

In Kombination mit leistungsfähigen Modellierungswerkzeugen ermöglicht dies die Erstellung einer komplexen, aber konsistenten Systemarchitektur über alle Domänen und die verschiedenen Phasen der Systementwicklung hinweg.

RFLP – ein MBSE-Ansatz

RFLP [10] ist einer der bekanntesten Ansätze zur Strukturierung von MBSE. Ursprünglich wurde er für die Entwicklung technischer *Produkte* entlang des V-Modells entwickelt.

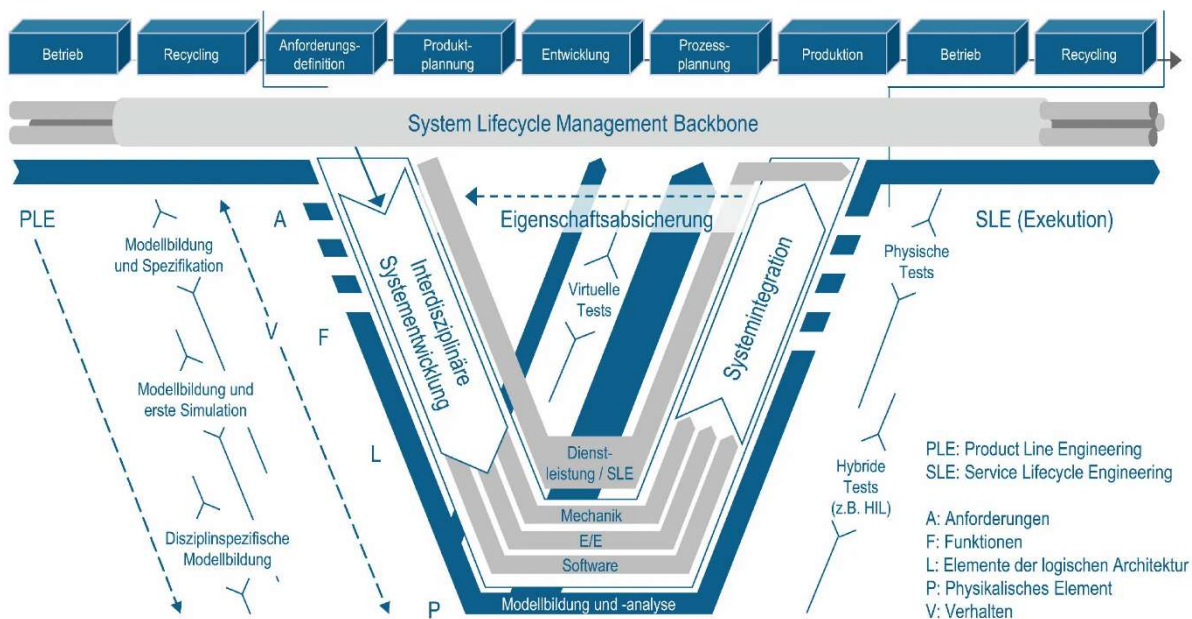


Abbildung 42: RFLP-ausgerichtetes V-Modell [11]

Dementsprechend entwickelten wir eine RFLP-basierte Methode zur Beschreibung der Architektur von *Simulation* als Gesamtsystem. Wie in Abbildung 42 dargestellt [11], differenziert RFLP die Ebenen

- der Anforderungen sowie der
- funktionalen,
- logischen und
- physikalischen Architektur und Modelle.

Anforderungen werden als Input für den Ansatz angenommen und z. B. von Kunden gestellt.

Die *funktionale Architektur* beschreibt die Architektur der Funktionen des Systems, die für die Erfüllung der Anforderungen nötig sind oder auch physikalische und strukturelle Notwendigkeiten repräsentieren. Im Allgemeinen können Funktionen hierarchisch unterteilt werden.

Die *logische Architektur* definiert Komponenten, die die Funktionen implementieren. Je nach Bedarf werden in der Regel Funktionen mit geringer Relevanz für die aktuelle Simulationaufgabe weggelassen oder vereinfacht. Häufig benötigt eine einzelne Funktion mehr als eine logische Komponente zur Implementation, und eine einzige logische Komponente unterstützt die Implementation mehrerer verschiedener Funktionen. Daraus ergeben sich *n:m*-Beziehungen. Innerhalb der logischen Architektur kann eine Unterebene zur Definition der technischen Basis der Komponenten und ihrer Schnittstellen abgegrenzt werden.

Die *physische Architektur* im Sinne von RFLP beschreibt das eigentliche, meist greifbare Produkt. In unserem Kontext übertragen wir diesen Begriff auf das *Simulationssystem* (Simulationsmodell, das für einen Simulationslauf bereit ist, ausgeführt von einem Simulatorekern) sowie auf die *Simulationaufgabe* im Rahmen eines *Simulationsprozesses*. Um diesen Unterschied zu verdeutlichen, nennen wir diese Ebene *konkret* statt physisch.

Die konkrete Architektur enthält also insbesondere die tatsächliche Anzahl von *instanziierten Objekten*, z. B. 8 konkrete Fußgänger-Instanzen desselben logischen Fußgänger-Modells. Wenn Modelle Parameter enthalten, muss die konkrete Architektur Regeln für deren Belegung bzw. Diskretisierung angeben. So werden parametrisierte Serien von Simulationsläufen für dieselbe Simulationaufgabe erzeugt, wie sie z. B. für Optimierungsanwendungen oder Kritikalitätsanalysen typisch sind.

Der RFLP-Ansatz wurde im Projekt unserer Kenntnis nach erstmals von der Entwicklung eines Produkts auf die Simulation als „Gesamt-Problemstellung“ übertragen.

Die Orientierungsmatrix der integrativen Architektur der Simulation

Eine erste Visualisierung dieses „Drei-Ebenen-Ansatzes“ zur Architekturbeschreibung verdeutlichte bereits die wesentliche Grundstruktur der integrativen Architektur der Simulation in einer Orientierungsmatrix (vgl. Abbildung 43). Dabei bilden die Domänen bzw. Architektur-Gegenstände die hierarchische Gliederung der gesamten Simulation in Spalten. Diese Matrix konnte bereits als Schablone zur Sortierung von Modellen, Formaten, Tools etc. dienen.

Architektur-Ebenen × Architektur-Gegenstände

(als Schablone zur Sortierung von Modellen, Formaten, Tools, ...)

	Simulation System								Simulation-based Engineering Task	
	Complete Simulation Model						Simulation Core		...	
	Traffic Participant				Traffic Environment				Sim. Setup	References → Model, Core
	Sensors	DF	Actuators	Dynamics	Execution	KPIs, ...
Functional Arch.										
Logical Arch.										
Physical/Concrete Arch.										

Abbildung 43: Erster Entwurf der Orientierungsmatrix

Die „Systems under Test“ (SuTs) der drei SUCs des Projekts verorten sich in unterschiedlichen Spalten der Schablone – von links nach rechts:

- Der SUC3 befasst sich mit der Absicherung von Komponenten des Traffic Participants. Sein SuT ist also in *einer* der Spalten unterhalb „Traffic Participant“ angesiedelt.
- Der SUC2 betrachtet einen kompletten Traffic Participant. Der SuT überdeckt daher den gesamten Bereich *aller* Spalten unterhalb „Traffic Participant“.

- Im SUC1 schließlich geht es weniger um einen einzelnen Traffic Participant, sondern um kritische Situationen, die sich im Allgemeinen aus der *Interaktion mehrerer Traffic Participants* ergeben. Sein Ergebnis sind daher kritische Szenarien. Sie befinden sich im Schema unterhalb "Traffic Environment".

Diesen Ansatz weiterverfolgend, gelangten wir im Projekt zu der Erkenntnis, dass sich in der Orientierungsmatrix auch Flüsse von Signalen, Informationen und Daten verorten lassen. Zusätzlich lassen sich die Bedarfe nach horizontaler und vertikaler Traceability sowie nach Glaubwürdigkeit (Credibility) darstellen:

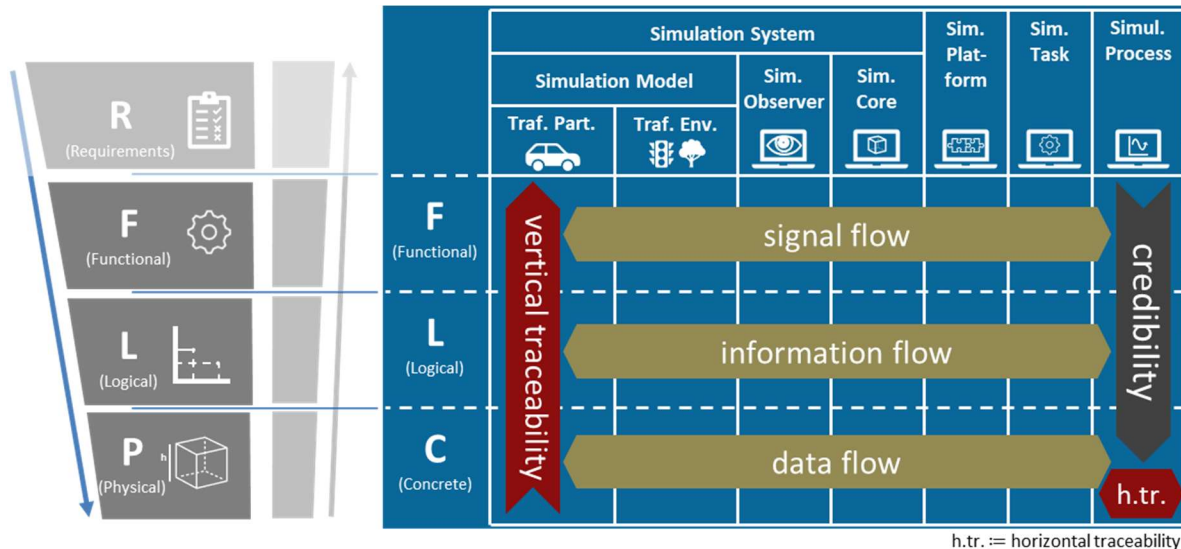


Abbildung 44: Endstand der Orientierungsmatrix

Die Orientierungsmatrix in dieser Gestalt

- adaptiert das RFLP-Muster aus MBSE an das Gesamt-Simulationssystem (eine Spalte pro Teilsystem/Domäne),
- ist Mittel zur Strukturierung und effizienten Kommunikation, (Jedes Feld enthält einen überschaubaren Teil der Architektur.)
- dient der Vollständigkeitsprüfung,
- stützt Informationsdurchgängigkeit und Passung der Schnittstellen „by design“, (Die zeilenweise Durchquerung erzeugt eine vollständige Liste der Schnittstellen.)
- veranschaulicht horizontale und vertikale Traceability,
- hilft Fehlerquellen zu reduzieren,
- um schließlich einen glaubwürdigen Simulationsprozess zu erreichen.

Im Bereich *Zusammenarbeit und Automatisierung* schafft die Orientierungsmatrix effiziente Kommunikation durch

- gemeinsame Sicht auf die Systemstruktur,
- gemeinsames Verständnis der Diskussionsebene
- und gemeinsames Wording.

Das verhindert Missverständnisse und sorgt für *Effizienz*. Gemeinsames Verständnis und stabile Schnittstellen sind Voraussetzungen für die *Automatisierung*.

SSP-Dateien, ausgeleitet aus der konkreten Ebene des Architekturmodells im EA, erleichtern das automatisierte Modell-Assembly.

Die Orientierungsmatrix unterstützt die *Projekt- und Prozessplanung* bzgl.:

- Abschätzung des Bedarfs an Modellen und der dafür benötigten Zeit.
- Konfigurationsmanagement gemäß ISO 10007 [12].
- Ressourcenplanung: Menschen, Software, Hardware.

Auch die Wiederverwendung von Modellen (Re-use) wird durch die Orientierungsmatrix gefördert: Stabile, nachhaltige und effiziente Schnittstellen sind Voraussetzung für die Wiederverwendung von Komponenten. Auch Synergieeffekte zwischen verschiedenen Simulationstypen können so genutzt werden.

Safety First for Automated Driving (sog. SaFAD-Paper)

Im April 2019 erschien der Report „Safety First for Automated Driving“, der von namhaften Unternehmen der Automotive- und IT-Branche gemeinsam erstellt worden war [13]. Er beschreibt unter anderem die funktionale Architektur eines realen hoch automatisierten Fahrzeugs, vgl. Abbildung 45. Aus diesem Papier entstand im Jahr darauf der *Report ISO/TR 4804:2020* [14]. Als Nachfolger ist *ISO/AWI TS 5083* [15]. bereits in Arbeit. Das SaFAD-Paper [13] samt Nachfolger stellte einen wichtigen Ankerpunkt dar, gegen den wir unsere Arbeit kontinuierlich abgeglichen haben. Es bestätigte uns in unserem RFLP-Ansatz, dass auch das Paper auf der funktionalen Ebene ansetzte. Der Vergleich der Architekturen des *realen* und des *simulierten* Fahrzeugs lieferte uns wichtige Erkenntnisse. Diese Untersuchungen fanden im oberen linken Feld der Orientierungsmatrix (Zeile *Functional*, Spalte *Traffic Participant*) statt.

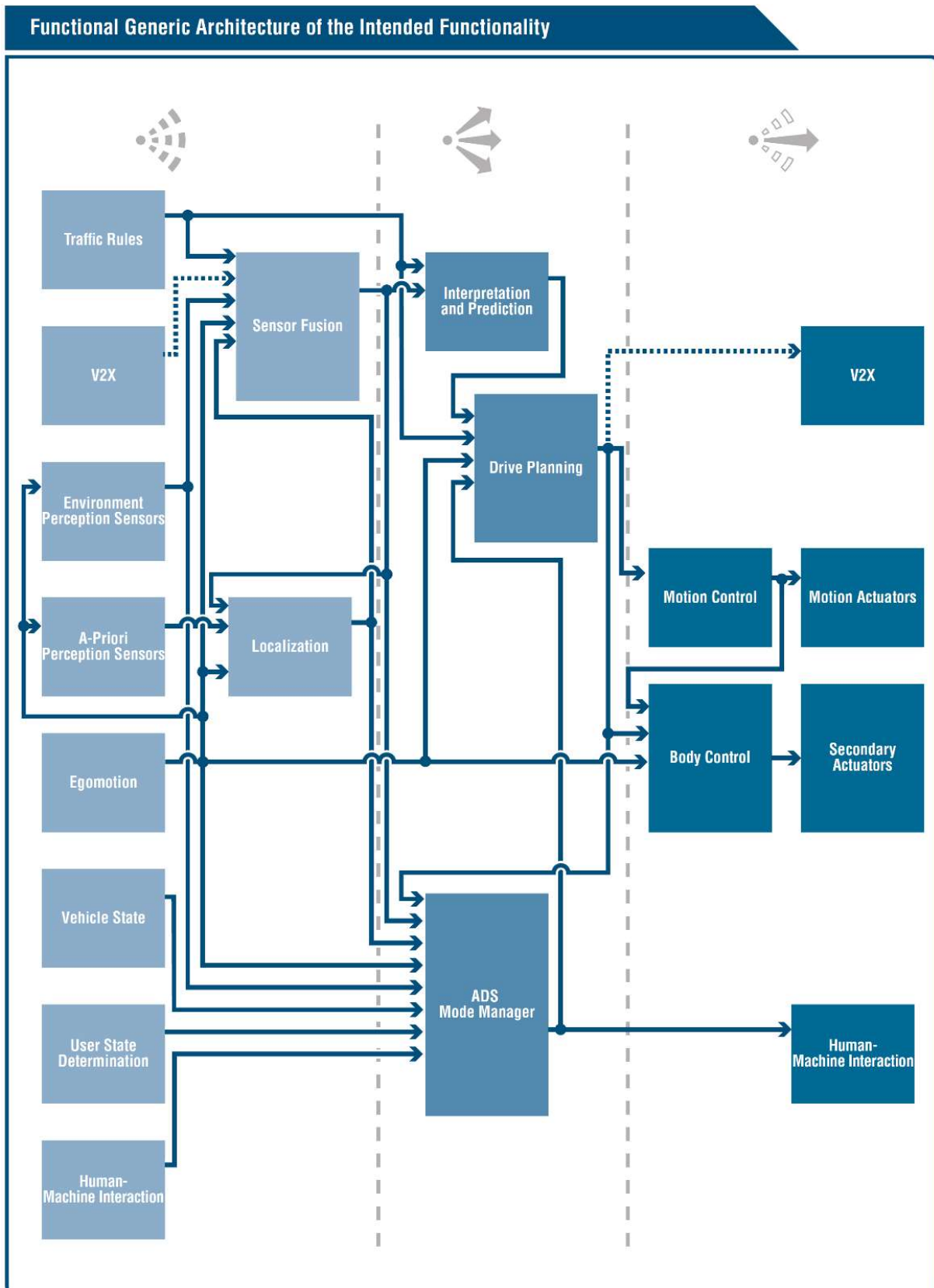


Abbildung 45: Funktionale Architektur eines realen Fahrzeugs (Fig. 26 des SaFAD-Papers [13])

UML-Modell der integrativen Architektur der Simulation mit Anwendung

Die Orientierungsmatrix kann, trotz der vielen oben aufgezählten Vorzüge, *nur eine Sicht von vielen* auf die Gesamtarchitektur bzw. das gesamte Architekturmodell sein. Für die tägliche Arbeit werden viele verschiedene Sichten für unterschiedliche Fragestellungen mit

unterschiedlichen Schwerpunkten und unterschiedlichen Detaillierungsgraden benötigt. Bereits zuvor haben wir beschrieben, dass wir uns im Projekt deshalb für den Einsatz eines geeigneten Tools (Enterprise Architect, kurz EA) entschieden haben. Mit diesem Tool erstellten wir ein UML-Modell der integrativen Architektur der Simulation und validierten es mit Fokus auf die SUCs 1 und 2 für die praktische Anwendung. Die gewonnenen Erkenntnisse von der allgemeinen Vorgehensweis bis hin zu effizienten „Klickreihenfolgen“ im Werkzeug sammelten wir in einer entsprechenden Modellierungsrichtlinie.

PROSTEP hat auch die Orientierungsmatrix in UML dargestellt, um die vielen einzelnen Architekturkomponenten des Gesamtmodells zu sortieren und eine Art *Inhaltsverzeichnis* zur Verfügung zu stellen:

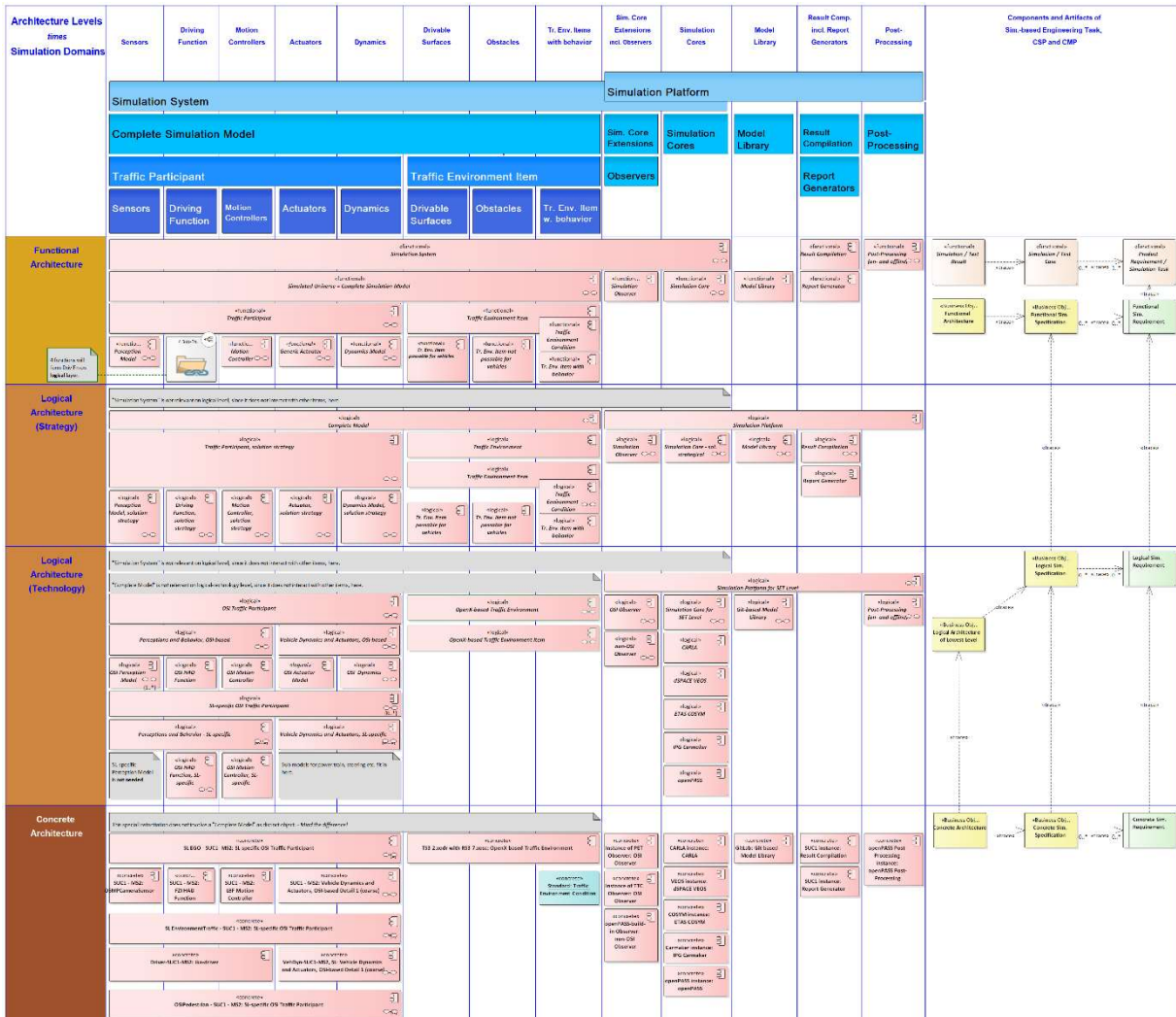


Abbildung 46: Orientierungsmatrix der wichtigsten Architektur-Komponenten, in UML dargestellt

Die vielfältigen *Beziehungen* zwischen diesen Komponenten sind hier der Übersicht halber nicht dargestellt. Dazu gehören vor allem

- *horizontale* Verknüpfungen, z. B.:
 - die Schnittstellen zwischen Traffic Participants, Simulation Core¹ und Observern,
 - oder übergeordnet

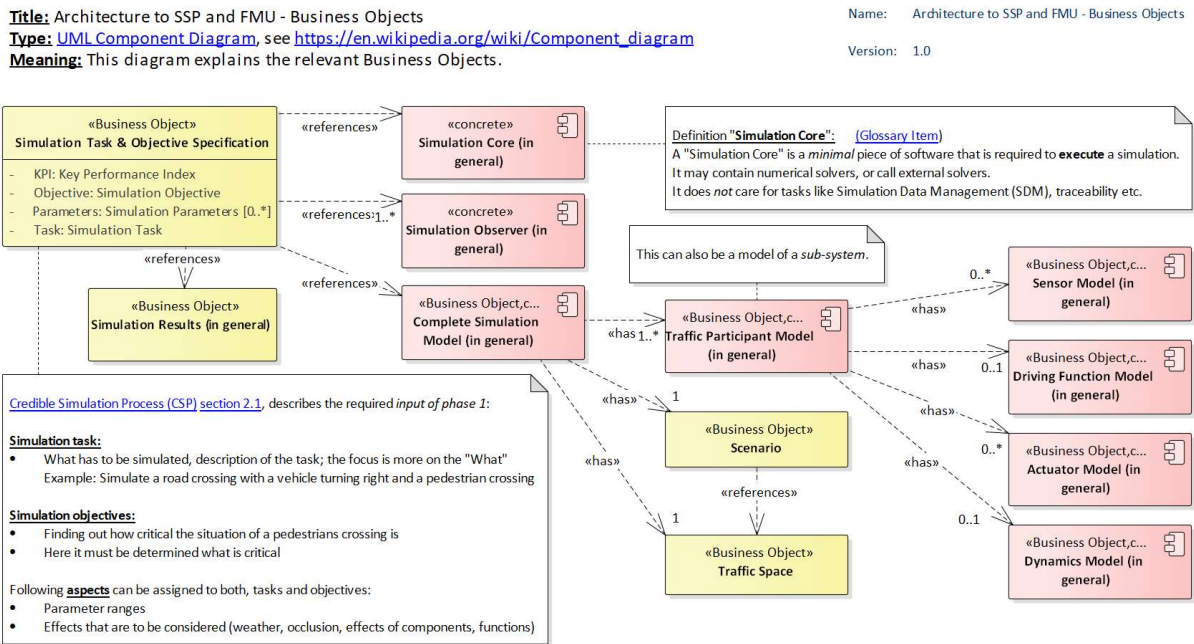
¹ Der *Simulatorkern* bzw. *Simulation Core* ist das minimale Stück Software, das zur Ausführung der Simulation ausreicht. Wird für die Simulation spezifische Hardware eingesetzt, z. B. HiL, so ist auch diese Hardware ein Teil des Simulatorkerns.

- die Schnittstellen zum Credible Simulation Process und dessen Artefakten wie Spezifikationen, Testbeschreibungen usw.
- vertikale Verknüpfungen, z. B. die Ableitung von Komponenten und deren Verknüpfungen aus der Architektur des realen Fahrzeugs, wie sie im Paper „Safety First for Automated Driving“ (SaFAD) [13] bzw. dem ISO TR 4804 beschrieben ist, über alle Ebenen der Simulationsarchitektur hinweg bis zur konkreten Architektur eines Simulationslaufs.

Das AP 2.1 hat aus der *funktionalen* Ebene des Gesamtmodells einen Liefergegenstand „Funktionale Beschreibung“ ausgeleitet.

Geschäftsobjekte der Simulation

Die Geschäftsobjekte (Business Objects), auf die man im Kontext der Simulation regelmäßig trifft, lassen sich in folgendem UML-Diagramm übersichtlich kategorisieren:



Die als FMU darstellbaren Objekte sind hier rosa hinterlegt, alle anderen gelb.

Abbildung 47: Geschäftsobjekte der Simulation

Unter den allgemeinen Simulationsergebnissen (Simulation Results in General) lassen sich *primäre*, *sekundäre* und *tertiäre* Ergebnisse unterscheiden. Die folgende Abbildung 48 setzt sie in Verbindung mit Phasen des CSP:

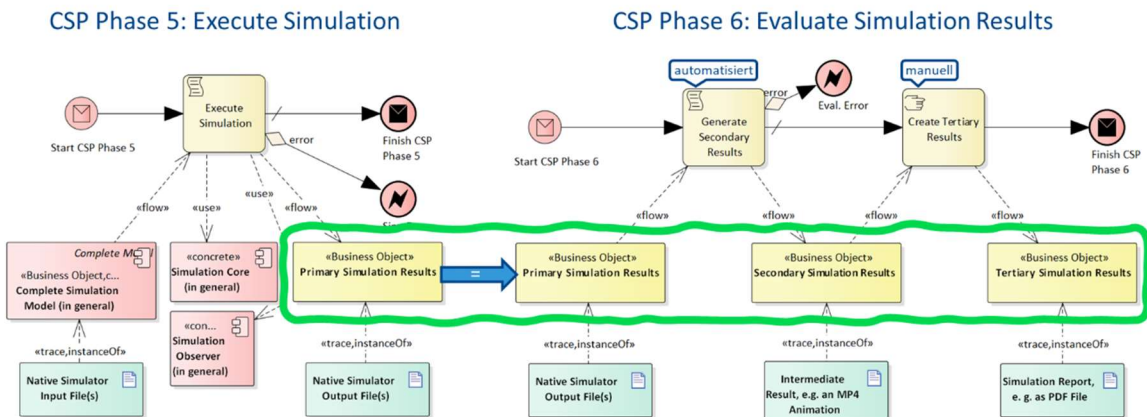


Abbildung 48: Primäre, sekundäre und tertiäre Simulationsergebnisse

Beschreibung der Methode am Beispiel einer Notbremsfunktion (Advanced Emergency Braking Systems - AEB)

Die methodische Ableitung der in Abbildung 46 nicht dargestellten vertikalen Verknüpfungen zwischen den Komponenten auf den unterschiedlichen Architektur-Ebenen wurde an einem vereinfachten Prinzip-Beispiel anhand einer hochautomatischen Notbremsfunktion (AEB) praktisch evaluiert. Das folgende Diagramm zeigt die funktionale Architektur des *realen* Fahrzeugs gemäß SaFAD [13] (Abbildung 45), in der die AEB-relevanten Komponenten blau markiert sind:

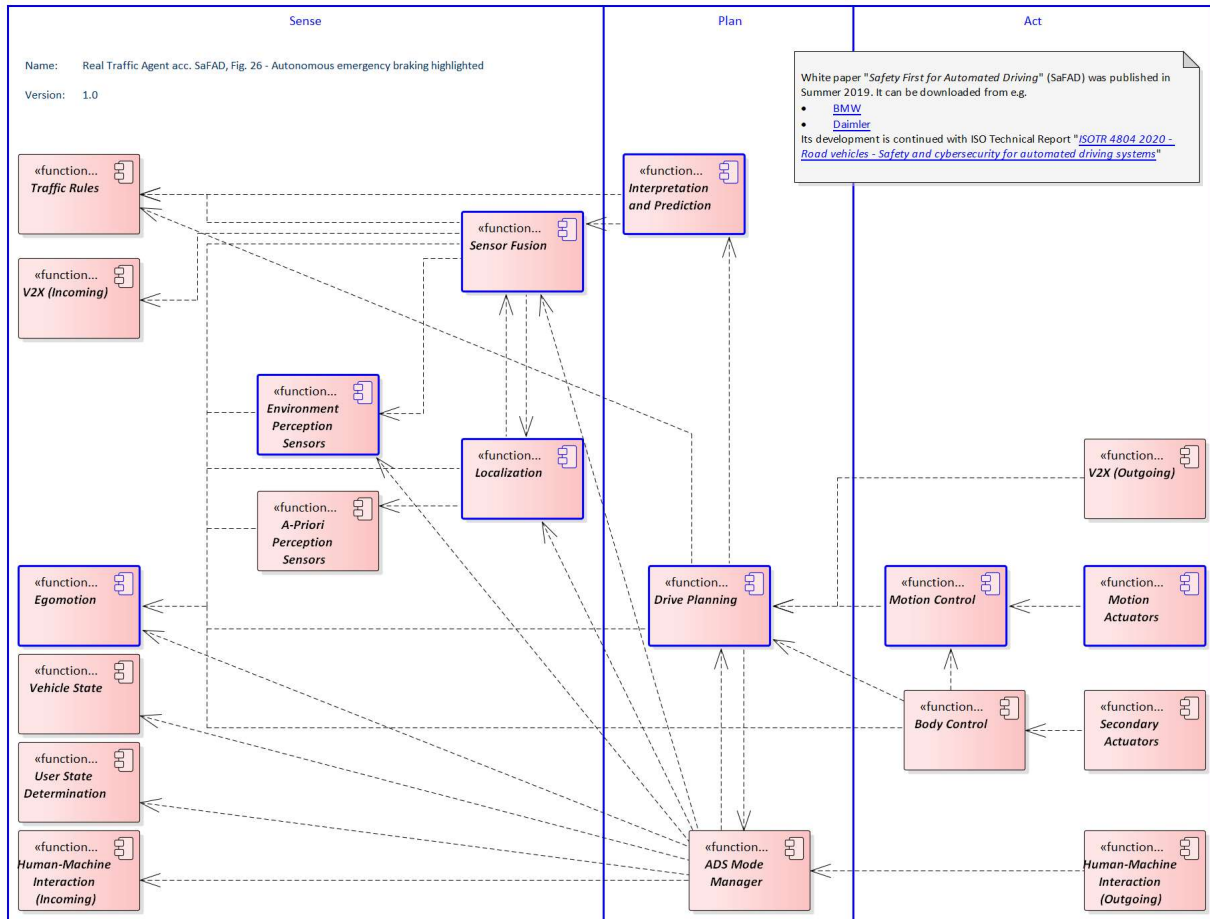


Abbildung 49: AEB-relevante Komponenten aus SaFAD [13]

Daraus leitet sich die oberste Architekturebene des simulierten Fahrzeugs wie folgt ab:

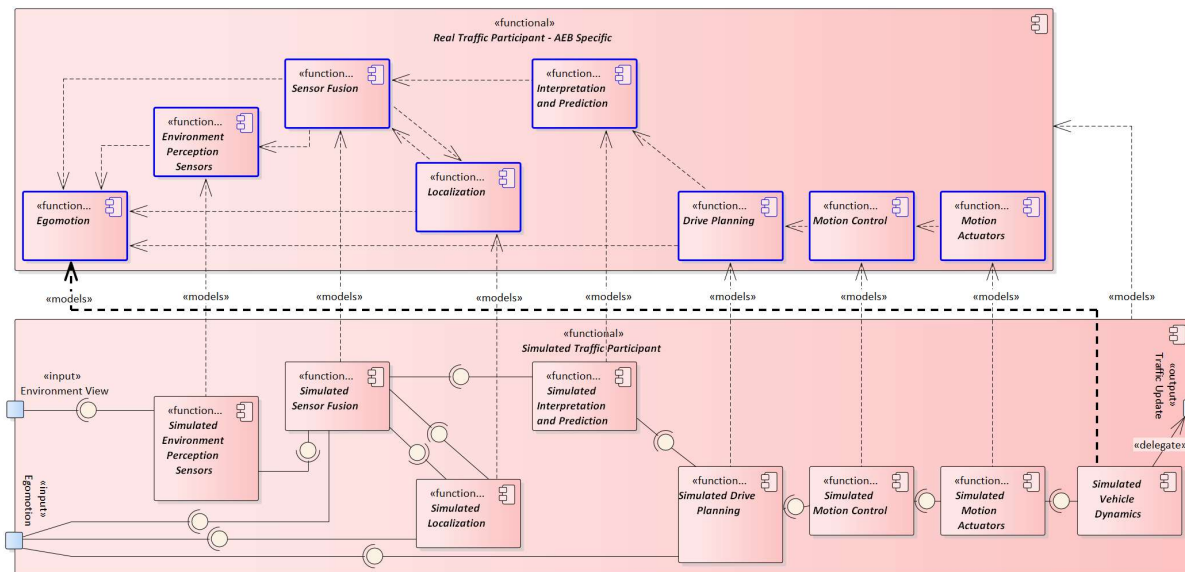


Abbildung 50: Oberste Ebene der AEB-Simulationsarchitektur

Die Abhängigkeiten (gestrichelte Pfeile in Abbildung 49) zwischen realen Komponenten werden in Abbildung 50 durch Assembly-Beziehungen zwischen simulierten Komponenten ersetzt. Simulierte Komponenten haben eine «models»-Abhängigkeit zu ihren realen Gegenstücken.

Die Komponente „Egomotion“ spielt in der SaFAD-Architektur [13] eine besondere Rolle: Sie „beschreibt den aktuellen Zustand des Fahrzeugs in Form von Gierrate, Längsbeschleunigung, Querbewegung und mehr. Weitere Werte, die den Fahrzustand beschreiben, können die Fahrzeuggeschwindigkeit oder der Schräglaufwinkel sein. Egomotion ist keine rein *interne* Komponente des Fahrzeugs, sondern eine Komponente, die das fahrdynamische Verhalten des Fahrzeugs *in seiner Umgebung* beschreibt. Die genannten physikalischen Messgrößen werden durch physikalische Naturgesetze bestimmt. In der Simulation ist die „Natur“ jedoch keine Instanz. Die Messgrößen können zu Beginn der Berechnung eines Zeitschrittes nicht als gegeben angenommen werden. Stattdessen müssen sie an dessen Ende explizit berechnet werden. Diese Funktion wird von der Komponente „Simulated Vehicle Dynamics“ bereitgestellt.

Der simulierte Verkehrsteilnehmer wird nun in den Kontext seiner Umgebung gesetzt. Auch dies folgt dem Vorbild der Realität, obwohl die Umgebung in SaFAD [13] nicht untersucht wird. Die Situation ist in der folgenden Abbildung dargestellt:

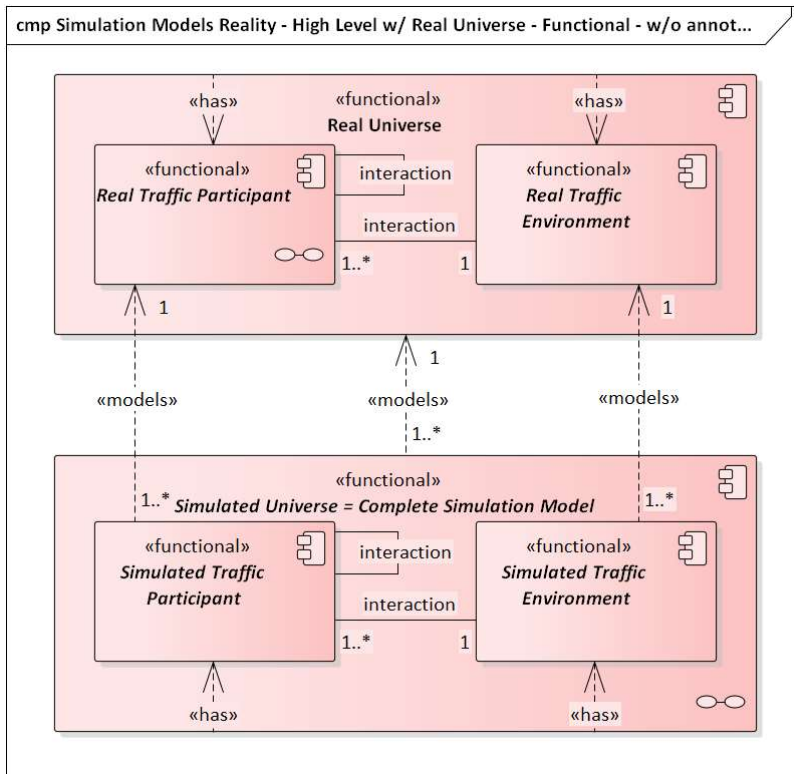


Abbildung 51: Reales und simuliertes Universum

Dementsprechend müssen die simulierten Sensoren mit Eingaben gefüttert werden, die aus der Ground Truth abgeleitet werden, aus anderen Verkehrsteilnehmern und der Umgebung. Die Ergebnisse der simulierten Fahrzeugdynamik müssen zurückgekoppelt werden, um die Ground Truth zu aktualisieren. In diesem Sinne unterscheidet sich das Ego-Fahrzeug nicht von anderen Verkehrsteilnehmern.

Es wird schnell klar, dass die wachsende Anzahl von Verkehrsteilnehmern, Elementen in der Verkehrsumgebung und deren komplexes Netzwerk von Interaktionen eine Orchestrierung erfordert. Außerdem müssen während und nach dem Simulationslauf Daten für die Nachbearbeitung extrahiert werden. Entsprechende Observer müssen mit den Simulationskomponenten verbunden werden. Dies ist der Punkt, an dem der Simulation Core² ins Spiel kommt. Insgesamt erhalten wir eine zweite Ebene der funktionalen Architektur der Simulation, die in der folgenden Abbildung des vollständigen Simulationssystems dargestellt ist:

² Siehe Fußnote 1.

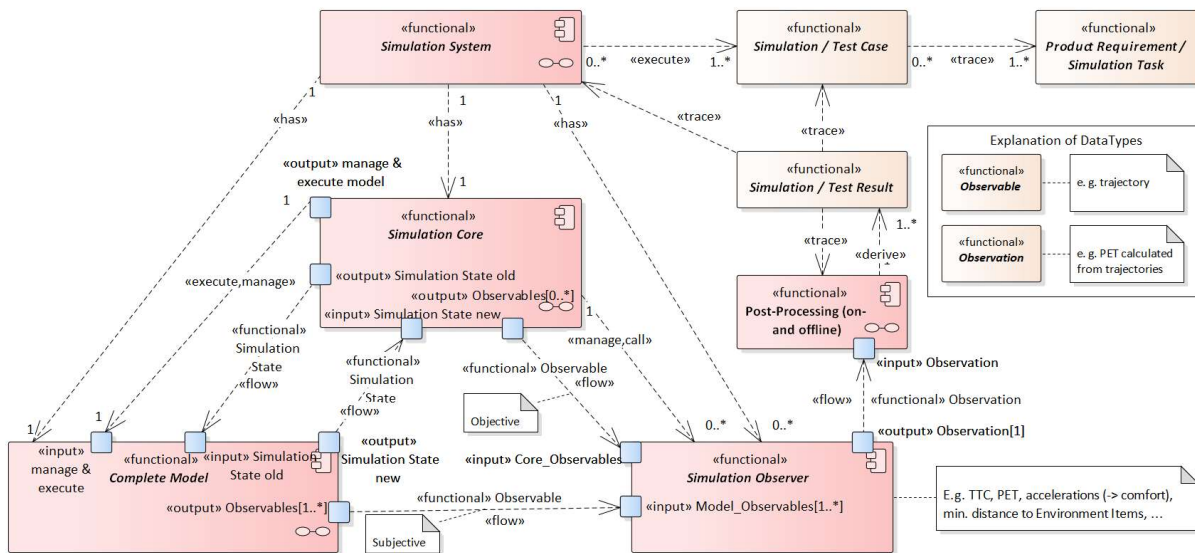


Abbildung 52: Funktionale Architektur des vollständigen Simulationssystems

Das Diagramm zeigt auch auf, wie das Simulationssystem mit dem Simulationsfall und dem Simulationsergebnis verknüpft ist.

Wesentlich für die Extraktion relevanter Daten aus einem Simulationslauf sind sog. *Observer*. Ein solcher *Observer* berechnet *Observations* aus *Observablen*. Eine *Observable* ist eine physikalische Größe, die (in der Realität) gemessen oder Variablen der Simulation entnommen werden kann, z. B. eine Trajektorie. Sie kann *objektiv* sein, d. h. der physikalischen Wahrheit entsprechen, oder *subjektiv*, d. h. der individuellen Interpretation der Realität durch einen Traffic Participant entsprechen. Eine *Observation* ist eine Größe, die aus einer oder mehreren *Observablen* *außerhalb* der eigentlichen Simulation *abgeleitet* wird, z. B. eine aus mehreren Trajektorien abgeleitete Time to Collision (TTC). Sie ist also in der Regel nicht direkt messbar.

In der *logischen* Architektur kann ein Observer fest im Core integriert sein, als FMU eingebunden werden oder separat nach Ende der Simulation aufgerufen werden.

Die interne Struktur des Gesamtmodells ist in der folgenden Abbildung 53 dargestellt. Ein wesentlicher Unterschied zu Abbildung 52 ist, dass es keine direkte Interaktion zwischen Verkehrsteilnehmern und Verkehrsumgebung mehr gibt. Stattdessen werden alle Verbindungen durch den Simulationskern geleitet. Da wir uns im Kontext einer dynamischen, in Zeitschritten organisierten Simulation befinden, erfüllt der Core mindestens folgende essentiellen Funktionen während der drei wesentlichen Phasen eines Simulationslaufs:

Initialisierung

- Laden, instanzieren und initialisieren derjenigen Komponenten des gesamten Simulationsmodells, die von Anfang an im Laufzeitsystem benötigt werden.
- Laden und initialisieren der Observer, sofern erforderlich.
- Herstellen aller Verbindungen zwischen Komponenteninstanzen und Observern, so dass später Informationen ausgetauscht werden können. (z. B. gem. einer SSP-Datei)

Nach der Initialisierung werden die Zeitschritte so lange wiederholt, bis ein Abbruchkriterium erfüllt wird.

Ausführung eines Zeitschrittes

- Einfügen neuer Traffic Participant in die Szene des anstehenden Zeitschritts, falls erforderlich. Dies umfasst die Initialisierung und Verbindung ihrer Komponenten.
- Versorgung der Modellkomponenten mit dem aktuellen Zustand der Simulation.

- Aufruf aller Modellkomponenten zur Berechnung ihrer Ergebnisse.
- Einsammeln aller dieser Einzelergebnisse.
- Aktualisierung und Konsolidierung eines konsistenten Gesamtzustands der Simulation entsprechend der Einzelergebnisse.
- Versorgung der Observer mit dem neuen Simulationszustand. Dies umfasst auch das Schreiben jeglicher Art von Ergebnis- oder Protokolldateien.
- Auswertung des Abbruchkriteriums und Beendigung der Simulation, falls erforderlich.
- Löschen von Traffic Participants aus der Szene, falls sie nicht mehr benötigt werden. Dies umfasst die Auflösung von Verbindungen und das Löschen von Komponenten.
- Bereitstellung des neuen Gesamtzustands der Simulation für die Neuverteilung im nächsten Zeitschritt.

Um Rechenzeit zu sparen, ist es üblich, an einzelne Komponenten nur den für sie jeweils relevanten Ausschnitt des Gesamtzustands weiterzuleiten.

Shut down

- Finalisierung und Schließen aller Arten von Ergebnis- und Protokolldateien.
- Auflösung aller Verbindungen und Löschen aller Komponenten aus dem Laufzeitsystem.

Diese Ebenen der funktionalen Architektur sind in der folgenden Abbildung dargestellt:

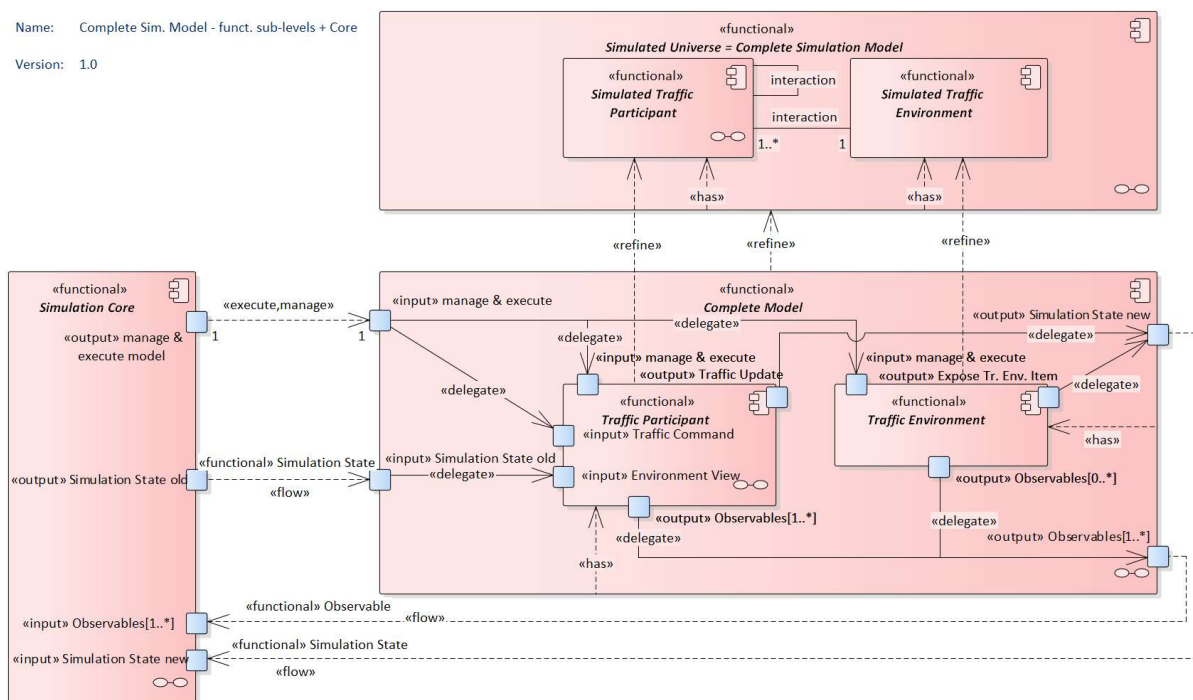


Abbildung 53: Beide Teil-Ebenen der funktionalen Architektur samt Simulation Core

Die obere Ebene in der Abbildung entspricht der unteren Ebene aus Abbildung 51.

Auf der zweiten Ebene der funktionalen Architektur erhält der SaFAD-konforme [13] Traffic Participant modifizierte Schnittstellen auf der Außenkante:

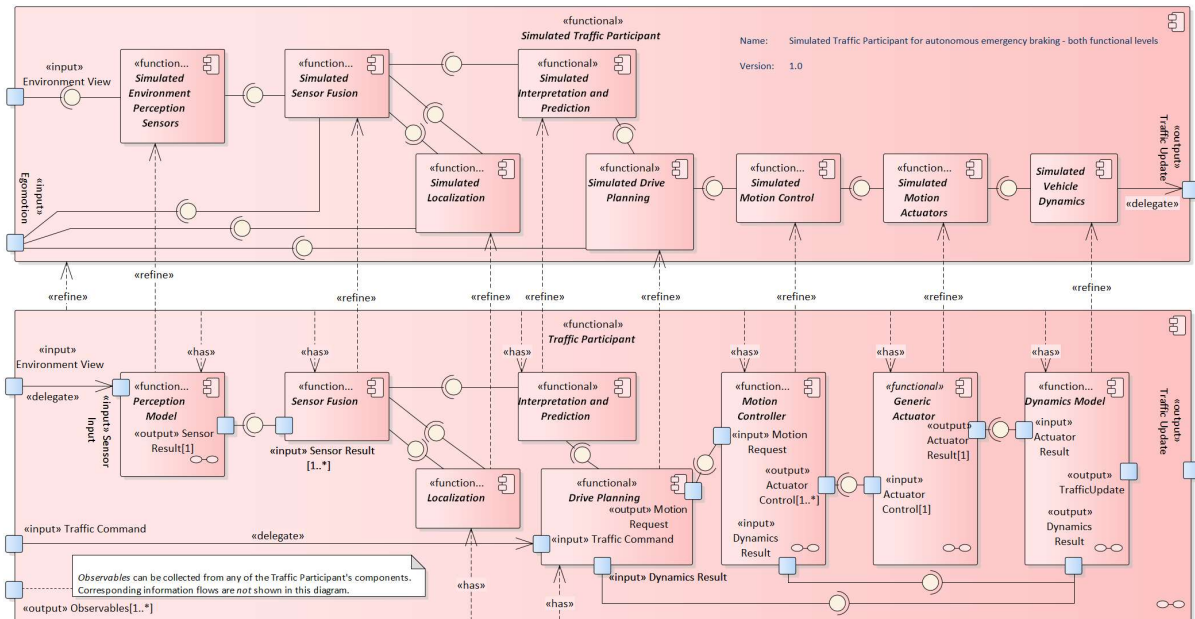


Abbildung 54: Beide Teil-Ebenen der funktionalen Architektur des Traffic Participants

Übergang zur logischen Architektur

Der funktionalen ist die logische Ebene der Architektur nachgelagert. Dort wird zunächst festgelegt, welche Funktionen von welchen logischen Komponenten der Simulation implementiert werden sollen. Beim Traffic Participant werden hier z. B. die vier Kernfunktionen Sensor Fusion, Localization, Interpretation and Prediction und Drive Planning zur Komponente Driving Function zusammengefasst. Der Zusatz solution strategy unterscheidet die logische Sub-Ebene der Lösungsstrategie von der nachfolgenden Sub-Ebene der technischen Umsetzung:

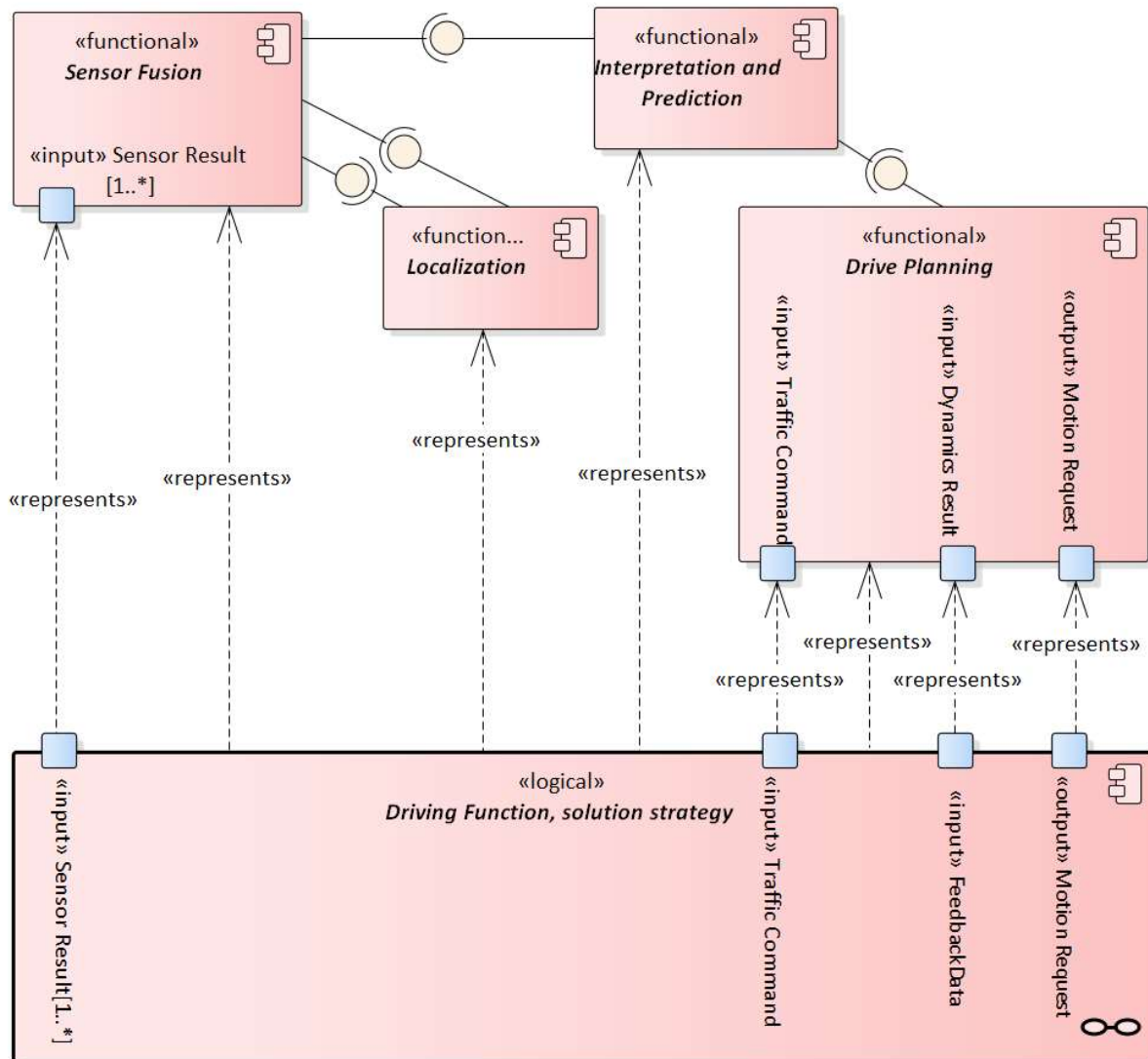


Abbildung 55: Aggregation von vier Funktionen zu einer logischen Komponente

Auf der logischen Sub-Ebene der technischen Umsetzung erfolgt dann die Festlegung auf zu verwendende Basis-Technologien, in unserem Fall FMI, SSP, OSI, OpenDRIVE und OpenSCENARIO.

Die Realität bietet eine unüberschaubare Vielfalt von Traffic Participants. Dazugehören nicht nur selbstfahrende PKW. Das Spektrum beginnt bei Tieren, die vor das Fahrzeug fliegen oder laufen, Bällen, die auf die Straße springen, Kindern, die womöglich hinterherrennen und reicht bis hin zu Schwerlasttransportern, Entsorgungsfahrzeugen, Bau-, Land- und Sondermaschinen. Es ist daher unbezahlbar bis unmöglich, all diese Traffic Participants vorab einzeln im Detail zu modellieren. Stattdessen hat das Projekt eine Methode aufgezeigt, wie sich die jeweils tatsächlich benötigten Traffic Participants effizient aus Einzelkomponenten konfigurieren lassen. Das geschieht in etwa so, wie auch ein Fahrzeughersteller entsprechend der jeweiligen Kundenwünsche ein individuelles Produkt herstellt, das oft vorher noch nicht in genau dieser Konfiguration produziert wurde. In beiden Fällen sind exakt definierte und leistungsfähige Schnittstellen die Voraussetzung für den Erfolg.

Diese Konfiguration erfolgt in mehreren Konfigurationsschritten gemäß folgender Abbildung:

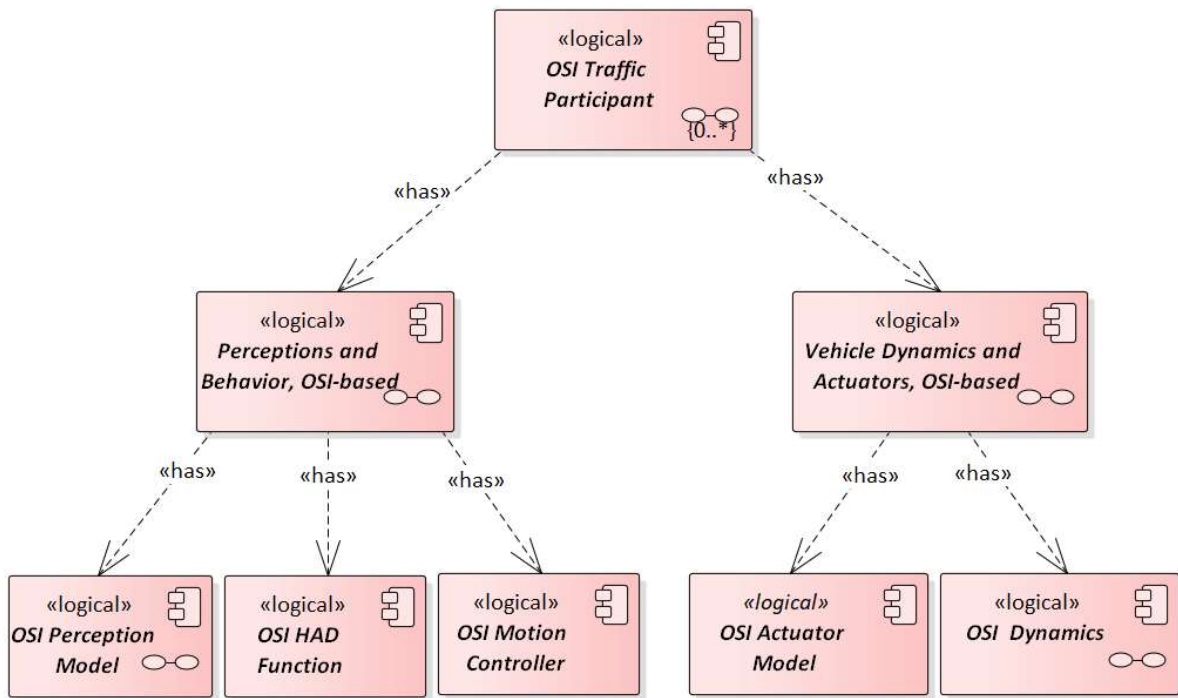


Abbildung 56: Konfigurationsgraph für den OSI-konformen Traffic Participant

Daher sieht der OSI-konforme Traffic Participant auf dem obersten dieser Konfigurationsschritte wie folgt aus:

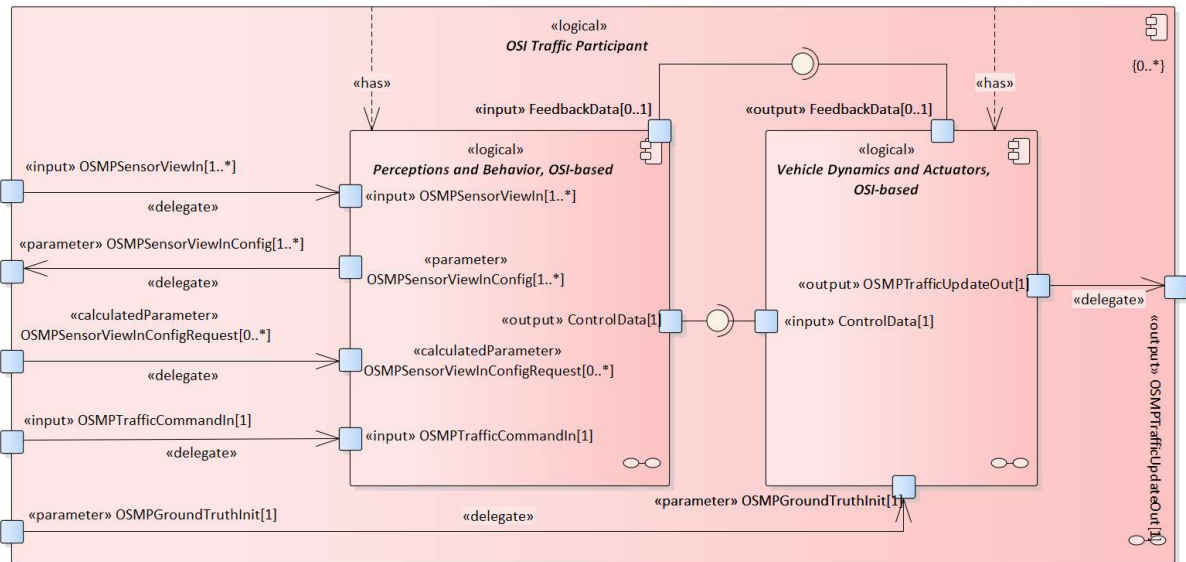


Abbildung 57: Konfigurierbarer OSI-konformer Traffic Participant

Für Perception and Behavior setzt sich das Schema wie folgt fort:

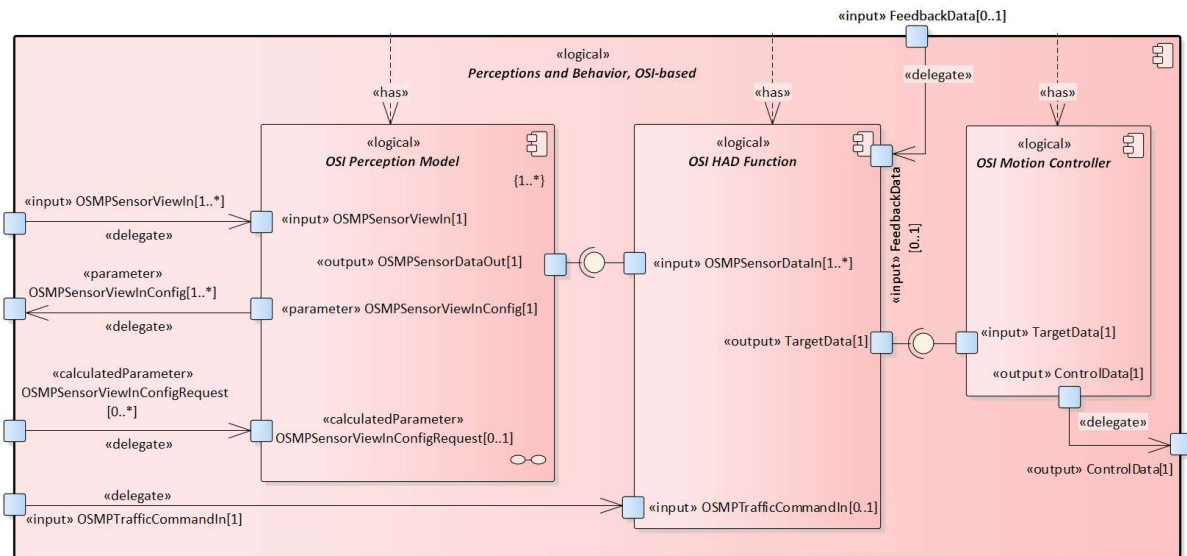


Abbildung 58: Konfigurierbare OSI-konforme Komponente Perceptions and Behavior

Diese Vorgehensweise ermöglicht zum einen die Instanziierung eines einfachen Modells des „Umgebungsverkehrs“ mit einem sehr rudimentären monolithischen Verhaltens- und Dynamikmodell und zum anderen die Instanziierung komplexerer Modelle durch Hinzufügen weiterer detaillierterer Komponenten.

Auf der nachfolgenden konkreten Ebene der Architektur kann der Verkehrsteilnehmer entweder als „Black Box“ mit nur den äußeren Schnittstellen aus Abbildung 58 oder als „Grey Box“ mit den internen Komponenten daraus instanziiert werden. Die Komponenten *Perception and Behavior* und *Vehicle Dynamics and Actuators* können dann wiederum entweder als „Black Box“ oder „Gray Box“ instanziiert werden. In diesem Fall zerfällt Komponente *Perception and Behavior* in *Perception*, *HAD Function* und *Motion Controller*.

Aus dieser einen Blaupause können nun die konkreten Architekturen aller Traffic Participants der Simulation aufgabe erstellt werden. Die dafür benötigten Teilmodelle können in einer Modellbibliothek vorgehalten werden. Durch die modellierten Beziehungen bleiben Traceability und Konsistenz jederzeit erhalten.

Fazit

Im Verlauf der beispielhaften Durchführung der Methode wurden die drei Hauptebenen (funktional, logisch & physikalisch) der Architektur bedarfsorientiert weiter ausdifferenziert. Repräsentationen der Geschäftsobjekte in Datenobjekten bzw. Dateien spezifischer Formate wurden beschrieben. Dies wird im folgenden Diagramm exemplarisch anhand der *logischen Architektur* dargestellt:

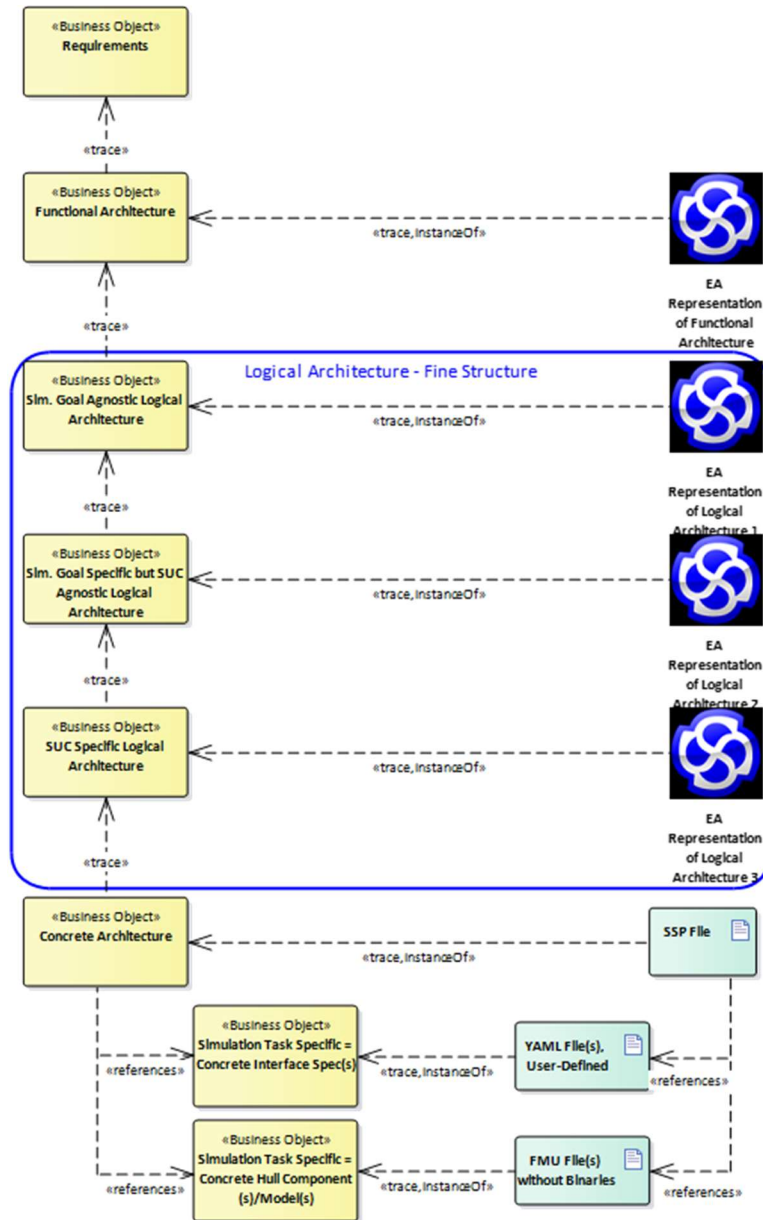


Abbildung 59: Ausdifferenzierung der logischen Architektur

Die rechte Spalte der Abbildung 59 stellt die Datenformate dar, die sich für die Arbeitsergebnisse (Artefakte) in der linken Spalte verwenden lassen.

Abschließend wollen wir noch den Faden aus dem vorigen Kapitel wiederaufgreifen und die funktionale Architektur eines realen Fahrzeugs gemäß des SaFAD-Papers [13] (vgl. Abbildung 45) im UML-Überblick der integrativen Architektur der Simulation verorten. Die nachfolgende Abbildung enthält dazu die Spalten der Orientierungsmatrix (Abbildung 46) aus technischen Gründen in einer permutierten Reihenfolge. Die SaFAD-Architektur [13] findet sich dabei am rechten Rand der zweiten Zeile von oben:

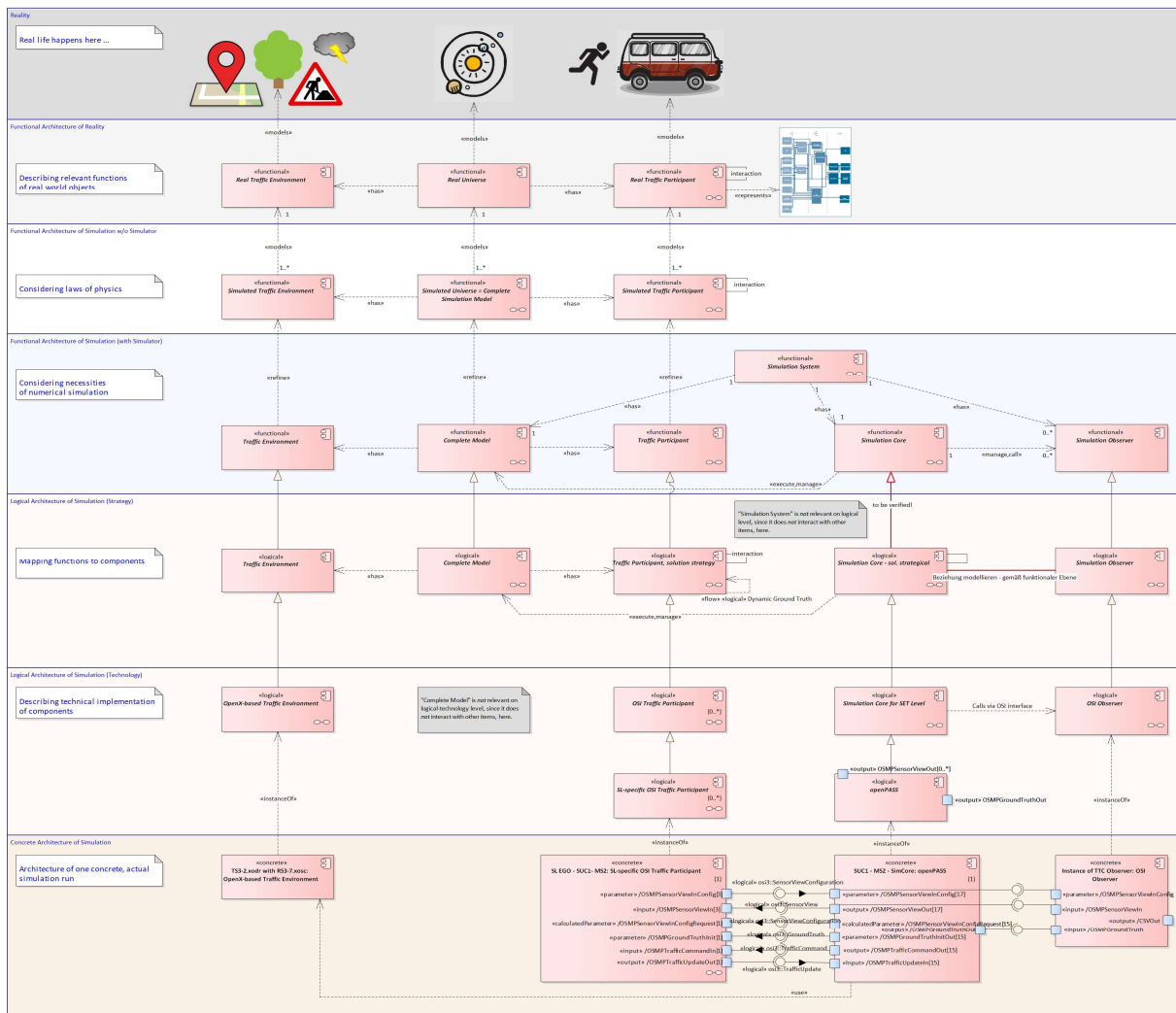


Abbildung 60: SaFAD-Paper [13] im Kontext der integrativen Architektur der Simulation

Diese Abbildung zeigt darüber hinaus die lückenlose Verknüpfung von Entitäten der realen Welt bis hinunter zu konkreten Simulationsmodellen. Alle Traceability-relevanten Entscheidungen lassen sich dabei einer spezifischen Architekturebene zuweisen, dort dokumentieren und in größtmöglichem Maße wiederverwenden. SSP-Dateien werden direkt von der untersten Ebene ausgeleitet.

Darüber hinaus wird klar ersichtlich, dass *Produktarchitektur* stets einen zwar wichtigen, aber nur vergleichsweise kleinen Teil der *Simulationsarchitektur* ausmacht.

Ausblick: Darstellung von Parametern in der Architektur

Eines der noch offenen Handlungsfelder betrifft die Modellierung und Darstellung von Parametern aller Art in der Architektur. Hier sind insbesondere folgende Fragen zu klären:

- Welche Arten von Parametern gibt es?
- Wo entstehen Parameter?
- Wie laufen sie durch die Architektur?
- Wie und wo gehen sie in den Simulationslauf ein?

CSP-Einbindung der Methode der integrativen Architektur der Simulation

Glaubwürdigkeit (Credibility) ist ein KO-Kriterium für den Einsatz von Simulation zur Produktfreigabe. Deshalb hat AP 3.1 der Weiterentwicklung des Credible Simulation Process (CSP) große Aufmerksamkeit gewidmet.

Architekturergebnisse stellen einen wichtigen Bestandteil diverser Spezifikationen dar.

Gemäß CSP müssen alle Artefakte, insbesondere Spezifikationen, auf glaubwürdige und nachvollziehbare Weise auseinander hervorgehen. Dementsprechend gilt dies erst recht für Architekturergebnisse.

Architekturergebnisse werden nicht in der Art „implementiert“ wie etwa Simulationsmodelle. Daher sind für sie nur die Phasen 1 bis 3 des CSP relevant.

Die Architekturergebnisse auf der *konkreten* RFLP-Ebene hängen von denen der *logischen* Ebene, und die wieder von den Ergebnissen der *funktionalen* Ebene ab.

Die typischen Zykluszeiten von Architektur-Änderungen unterscheiden sich erheblich zwischen diesen Ebenen:

- funktional Jahre
- logisch Monate
- konkret Tage,
 bei bloßer Parameteränderung ggf. auch nur Minuten.
 (Einbauposition der Kamera, Geschwindigkeit des Radfahrers, ...)

Deshalb ist es vernünftig, pro Architekturebene eine eigene Instanz des CSP vorzusehen.

Die nachfolgende Abbildung 61 stellt grafisch dar, wie die Ergebnisse dieser Instanzen in nachgeordnete Instanzen und letztlich in den CSP-konformen Simulationsprozess eingehen:

- *Funktionale* und *logische* Architektur gehen in die *Requirements Spec.* des CSP ein.
- Die *konkrete* Architektur geht in die *Design Specification* des CSP ein.

Abbildung 62 erweitert die Darstellung auf Details zu den CSP-Phasen 4 bis 6.

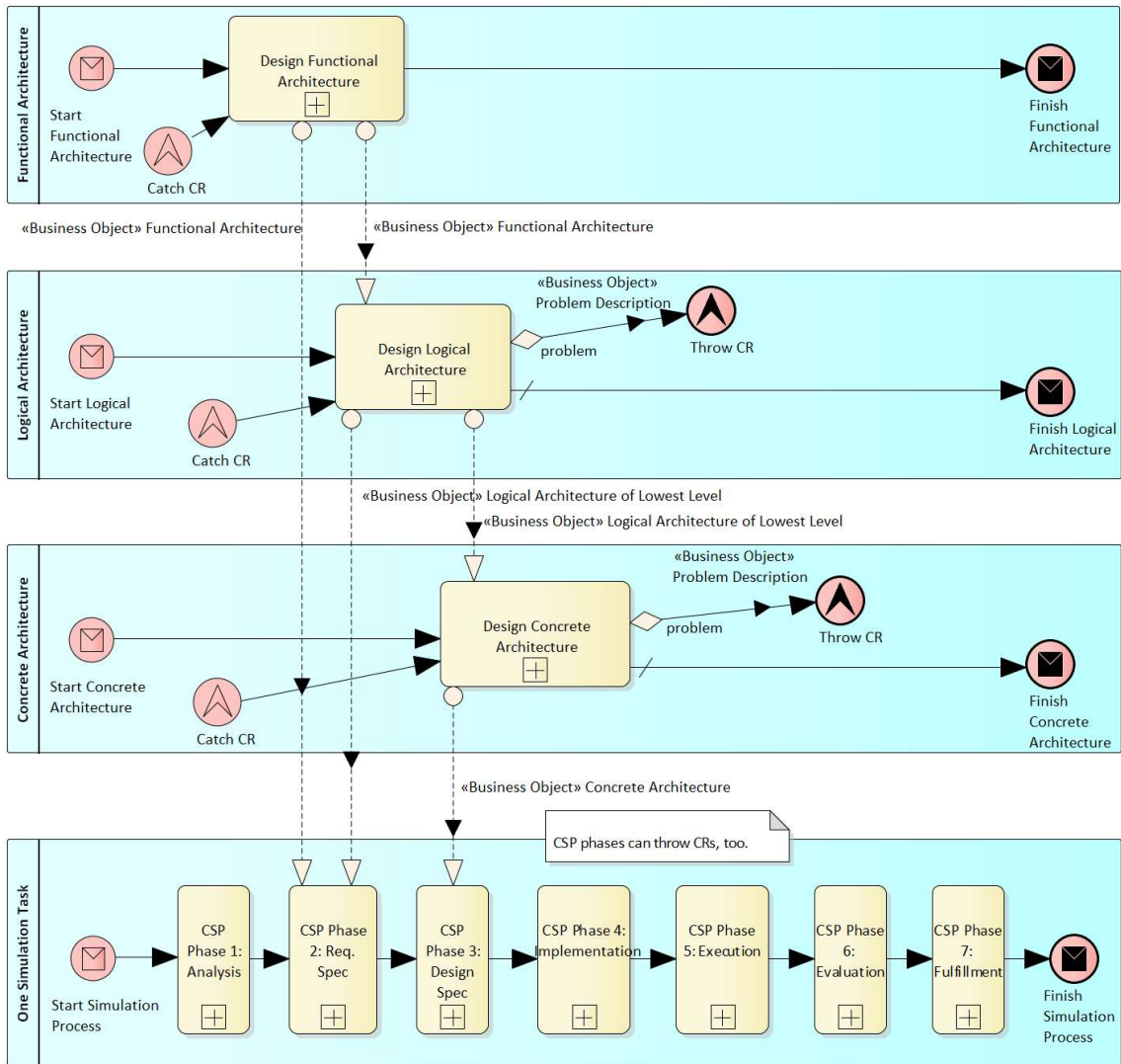


Abbildung 61: Zusammenspiel integrative Architektur und CSP

Title: Workflow: Architecture to SSP and FMU
Type: BPMN 2.0 Collaboration Diagram
 See https://en.wikipedia.org/wiki/Business_Process_Model_and_Notation
Meaning: Deriving the architecture of a specific simulation (task) involves several levels of abstraction, processes, tools, business objects/artifacts and their formats. For better overview, not all message flows are depicted, her. See [Message Flow Diagram](#) for those details.

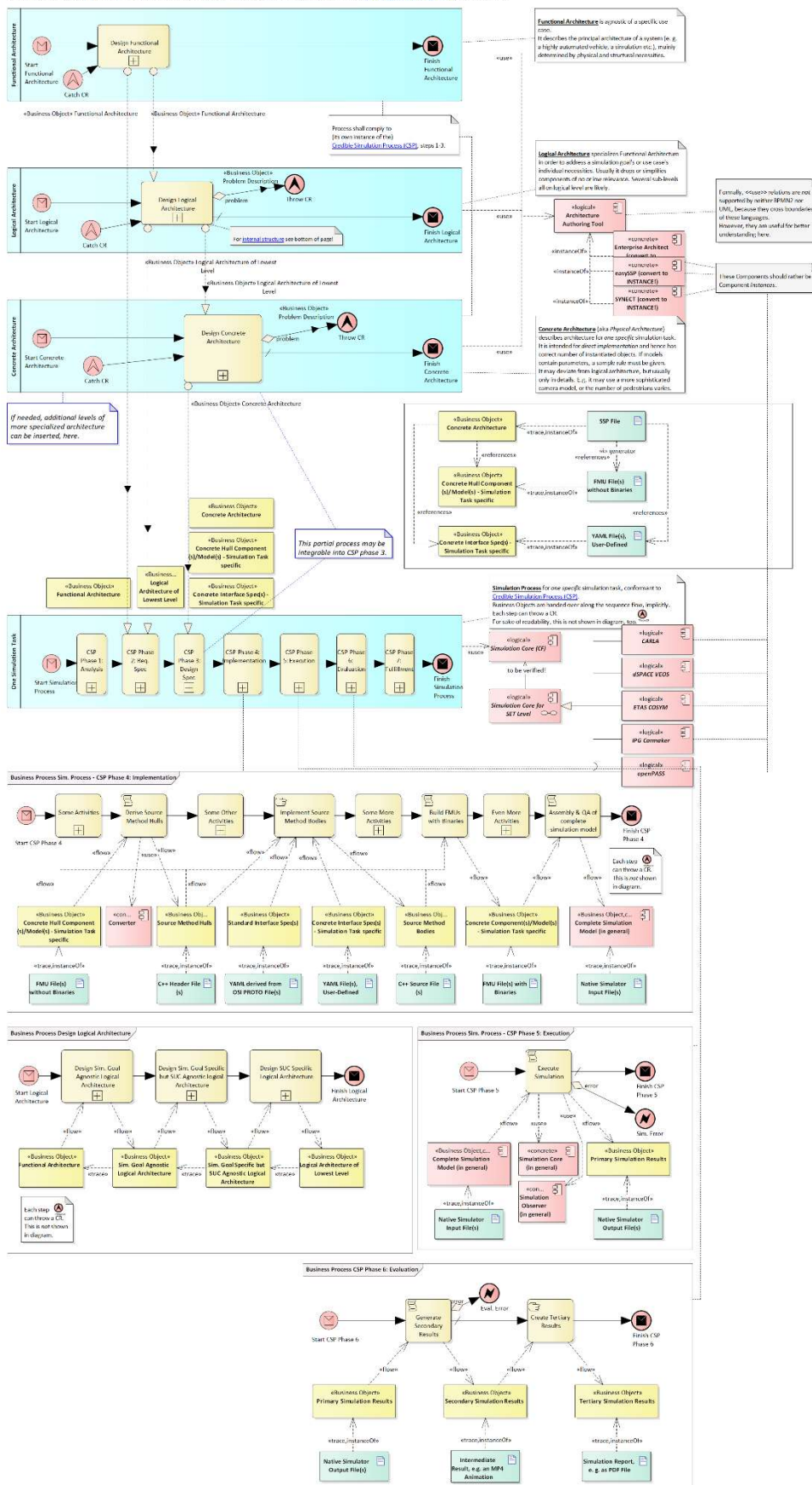


Abbildung 62: CSP-konforme Ableitung konkreter Architekturen aus abstrakteren Architekturen

Anwendungsfälle (Use Cases) und Motivation für die Architekturarbeit

Selbstverständlich muss auch für Architekturarbeiten eine Kosten-Nutzen-Betrachtung erfolgen. Für ein so abstraktes Thema kann dies aber typischerweise nicht in exakten Beträgen geschehen. Vielmehr müssen wichtige Anwendungsfälle und Motivationen identifiziert werden. Einige davon wurden bereits in der Einleitung des Architektur-Kapitels aufgeführt:

- Synergien nutzen.
- Orientierung bieten.
- Kommunikation erleichtern und Missverständnissen vorbeugen.
- Redundante Arbeiten sowie Fehlerquellen vermeiden.
- Direkte Unterstützung des Modellaufbaus in einer Simulationssoftware.
- Nachweis der Glaubwürdigkeit.
- Unterstützung der Planung und Koordination der Entwicklungsarbeiten.

Hinzu kommt der Aspekt der *Kostenoptimierung von Software-Lizenzen bis hin zu Zertifizierungen*. Hierzu wurde bereits früh im Projekt mit Fokus auf die Tool-Architektur die Frage identifiziert, wie weit sich eine *SUC-übergreifende, einheitliche Tool-Landschaft* definieren lässt. Die Frage erweiterte sich dann auch auf die Landschaften der Modelle und der Test- bzw. Simulationsfälle.

Je mehr dieser Architektur allgemein verwendbar ist, umso *ökonomischer* lassen sich Software-Lizenzen einsetzen, umso *weniger Overhead* entsteht durch zusätzliche Schnittstellen, und umso *kostengünstiger* fällt eine Zertifizierung einer Tool-Architektur für Produktfreigaben aus.

Die Ausrichtung von Arbeitsergebnissen auf der richtigen Architekturebene fördert deren Wiederverwendbarkeit und trägt so ebenfalls zur Kostenminimierung bei.

Architekturgebundener Einsatz von SSP: Motivation und Umsetzung

Der Standard *Functional Mockup Interface (FMI)* hat sich bereits über viele Jahre weiterentwickelt und bewährt. Er dient dazu, die Schnittstelle eines Verhaltensmodells zu beschreiben.

Den *Zusammenbau*, *ibes.* die *Verschaltung* solcher Modelle zu größeren Einheiten kann er jedoch nicht beschreiben. Diese Lücke schließt der Standard *System Structure and Parameterization (SSP)* erst seit März 2019, also im Wesentlichen zeitgleich zum Beginn des Projektes SET Level.

Daher war es ein Ziel des Projektes, diesen jungen Standard zu evaluieren.

Dementsprechend wurde von AP 4.1 gemeinsam mit AP 2.1 ein „Proof of Concept“ (PoC) entwickelt zur Darstellung eines durchgängigen, toolgestützten, CSP-konformen Prozesses von den höchsten, auf funktionaler Ebene abstrahierten Modellarchitekturen bis hin zu Source-Code-Rümpfen für z. B. Sensor-Modelle und deren Verschaltung mittels SSP.

Die theoretische Vorgehensweise dabei von der funktionalen bis zur konkreten Architekturebene wurde bereits in den vorangehenden Kapiteln beschrieben. Die *organisatorische*

Umsetzung des Ansatzes in der Praxis unter Einsatz einer Modell-Bibliothek ist in folgendem Diagramm dargestellt:

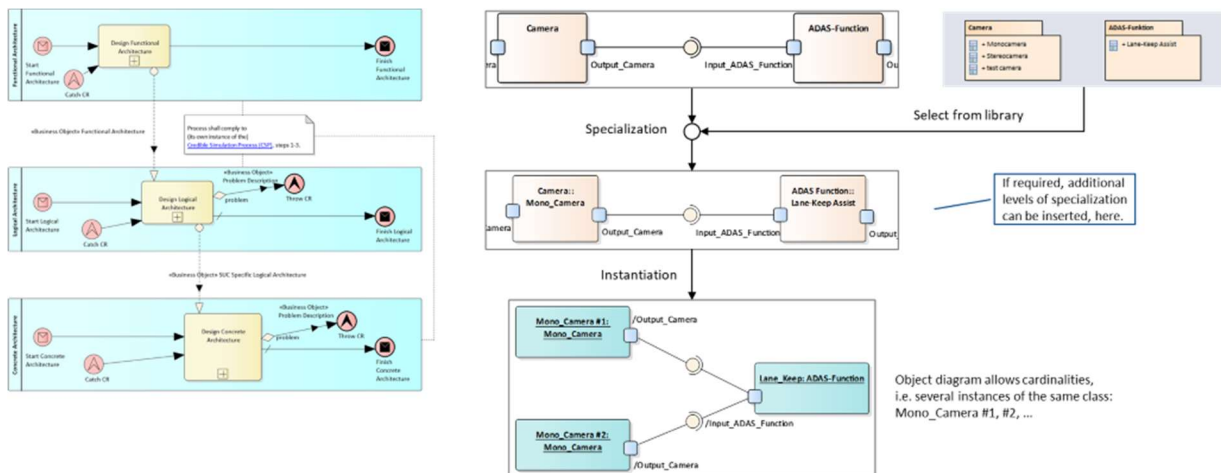


Abbildung 63: Organisatorische Umsetzung der 3-Ebenen-Architektur

Im Folgenden wird die Weiterverarbeitung der aus dem Enterprise Architekt im XMI-Format abgeleiteten konkreten Architekturen betrachtet.

ZF hat ein Werkzeug beigesteuert, das diese XMI-Dateien zur Weiterverarbeitung in anderen Simulationswerkzeugen ins Format SSP konvertieren kann. Dort kann dann z. B. nach folgendem Schema weitergearbeitet werden:

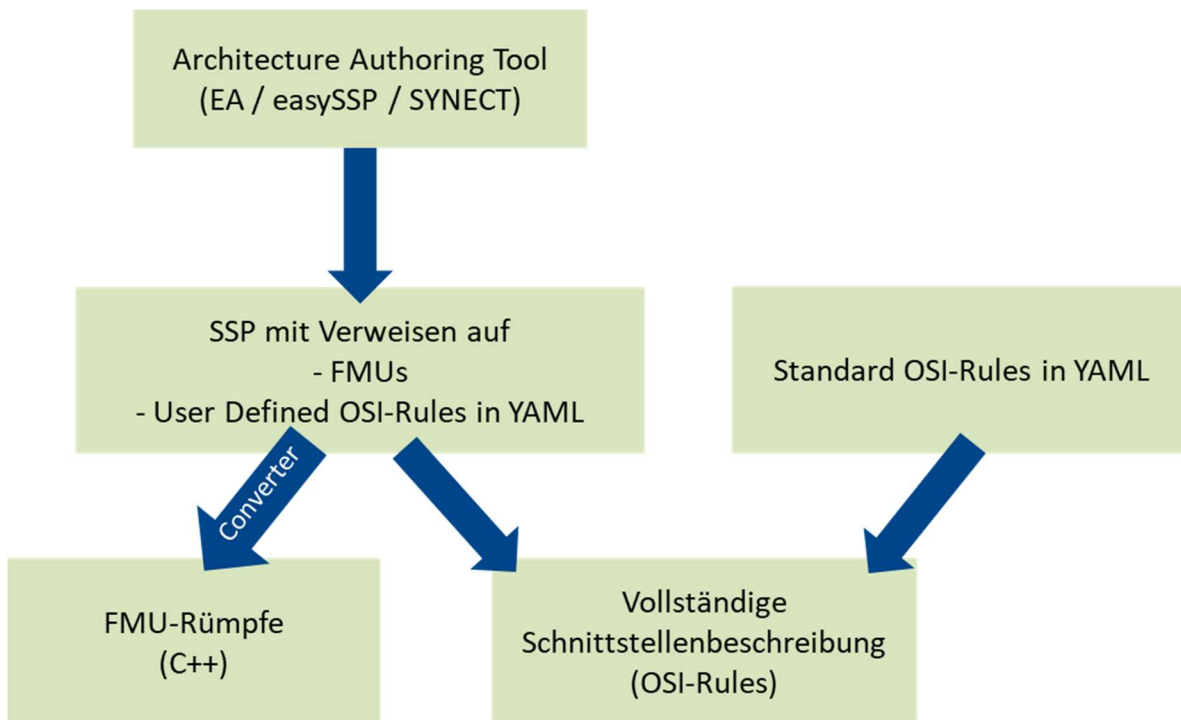


Abbildung 64: Ableitung von Modell-Rümpfen aus SSP-Dateien

Der entsprechende Prozess wurde an zunehmend komplexen Beispielen validiert.

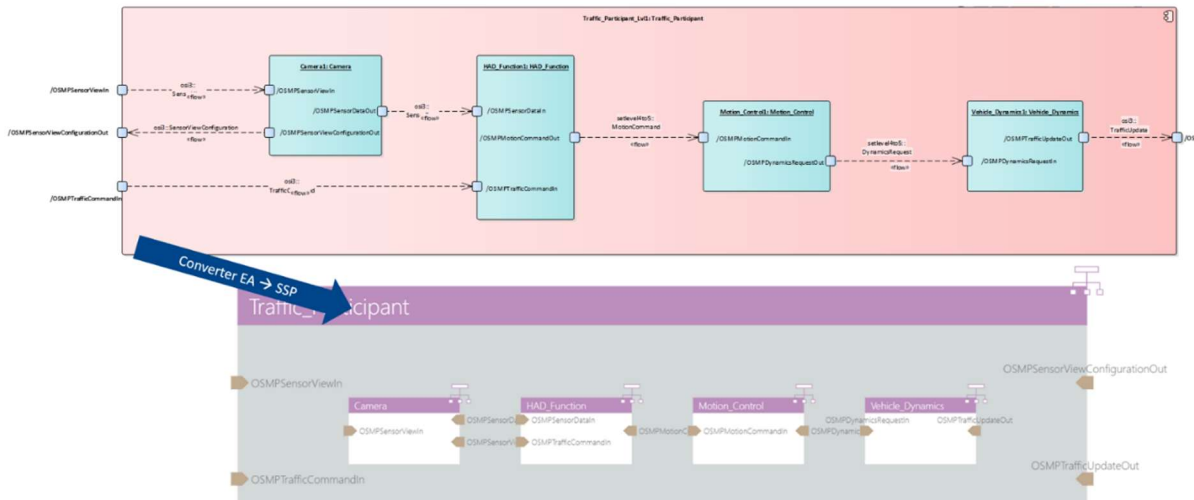


Abbildung 65: Erster Versuch zur Übertragung von Architektur nach SSP und weiter in andere Tools
Eine Validierung anhand eines komplexeren Beispiels folgte nach:

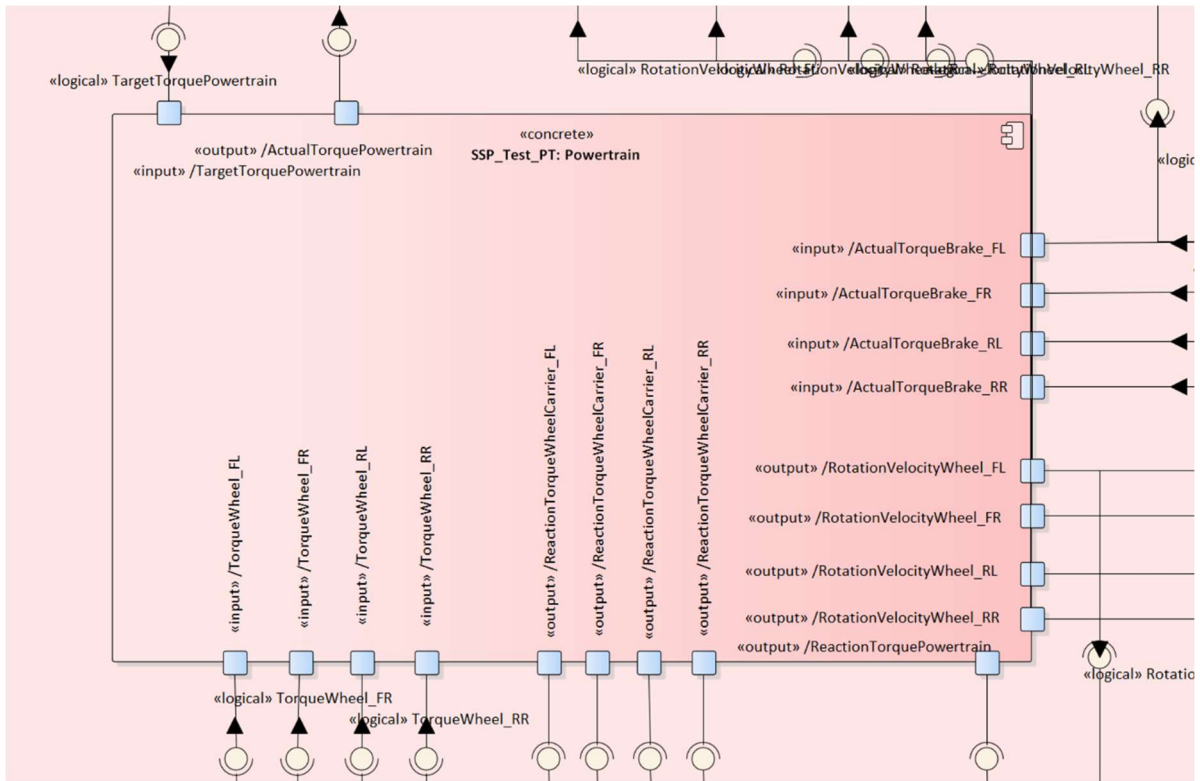


Abbildung 66: Darstellung des Powertrain-Blocks aus „Vehicledynamic and Actuators Model Detail 2“

Daraus wurde eine SSP-Datei erstellt und erfolgreich ins Tool easySSP importiert, wie in Abbildung 67 zusehen:

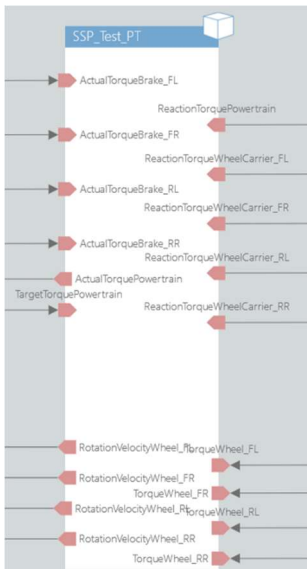


Abbildung 67: Graphische Darstellung des SSP zum oben abgebildeten Powertrain-Blocks

Zum Meilenstein 3 schließlich wurde gemeinsam mit AP 4.2 die konkrete Architektur eines kompletten Simulationslaufs von SUC1 betrachtet. Die folgende Abbildung stellt sie auf höchster Ebene (Simulatorkern (sog. Core) mit Verkehrsteilnehmern und Umgebung) dar:

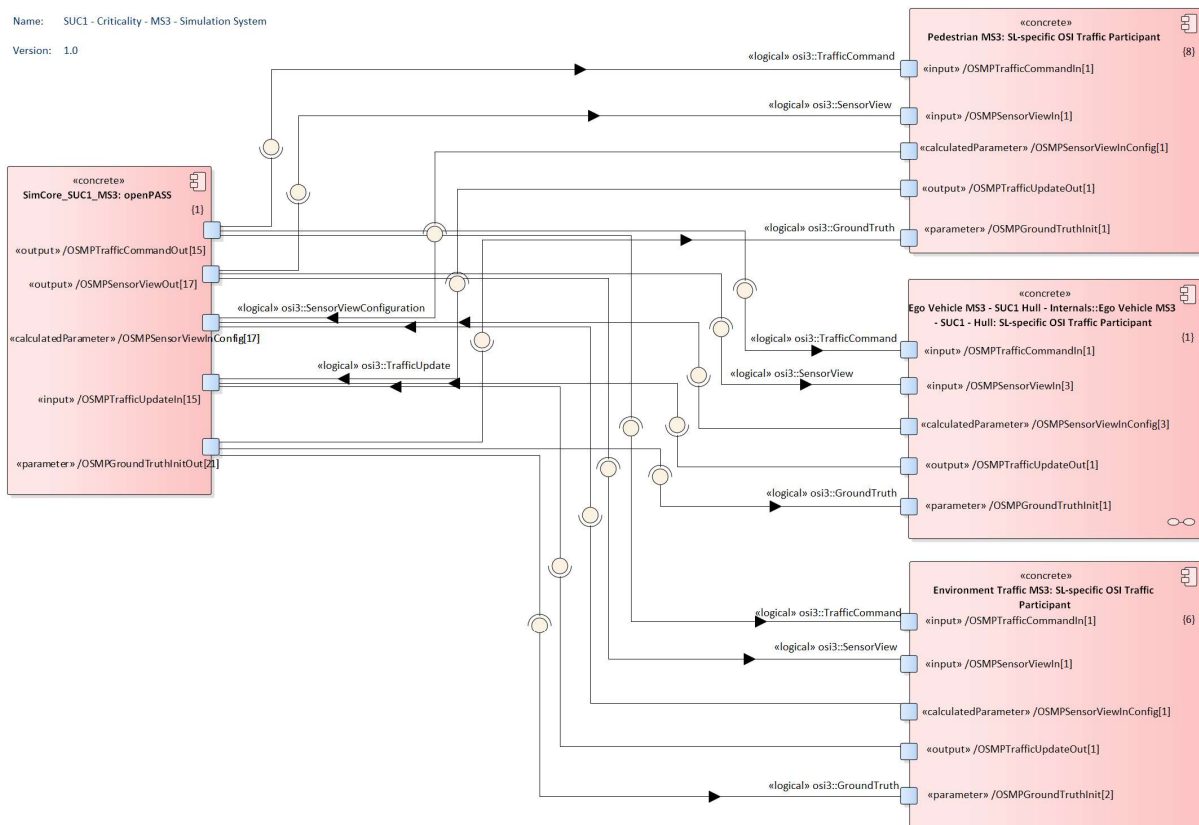


Abbildung 68: Konkrete SUC1-MS3 Architektur, Gesamtsicht

Die inneren Strukturen der Verkehrsteilnehmer und des Cores wurden in weiteren Diagrammen aufgelöst. Für das *Ego Vehicle* sieht dies wie folgt aus:

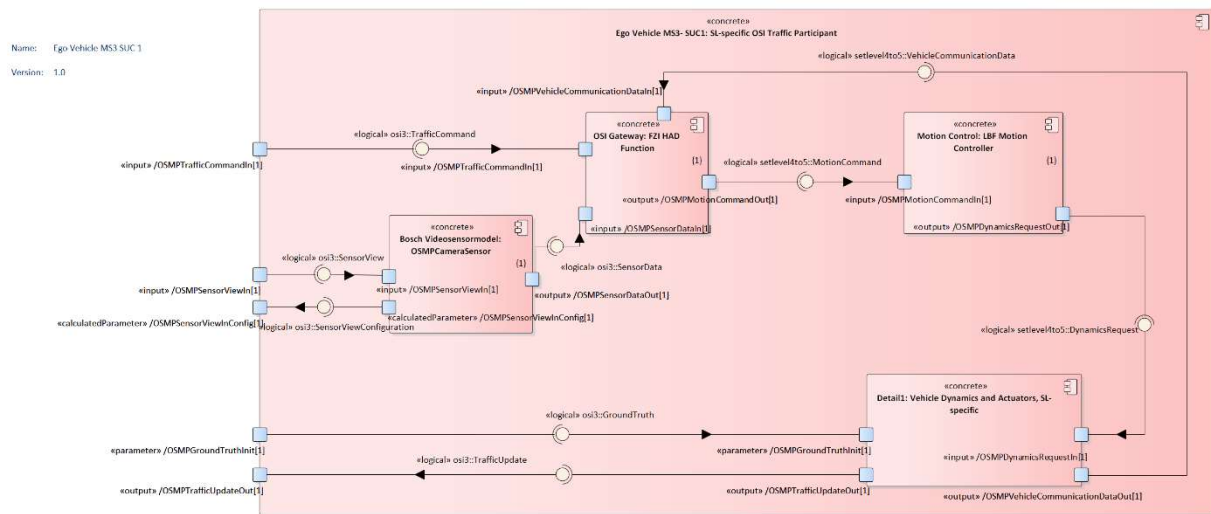


Abbildung 69: SUC1-MS3 Architektur, Sicht aufs Ego Vehicle

Dieses Diagramm wurde via SSP nach easySSP übertragen und dort korrekt interpretiert:

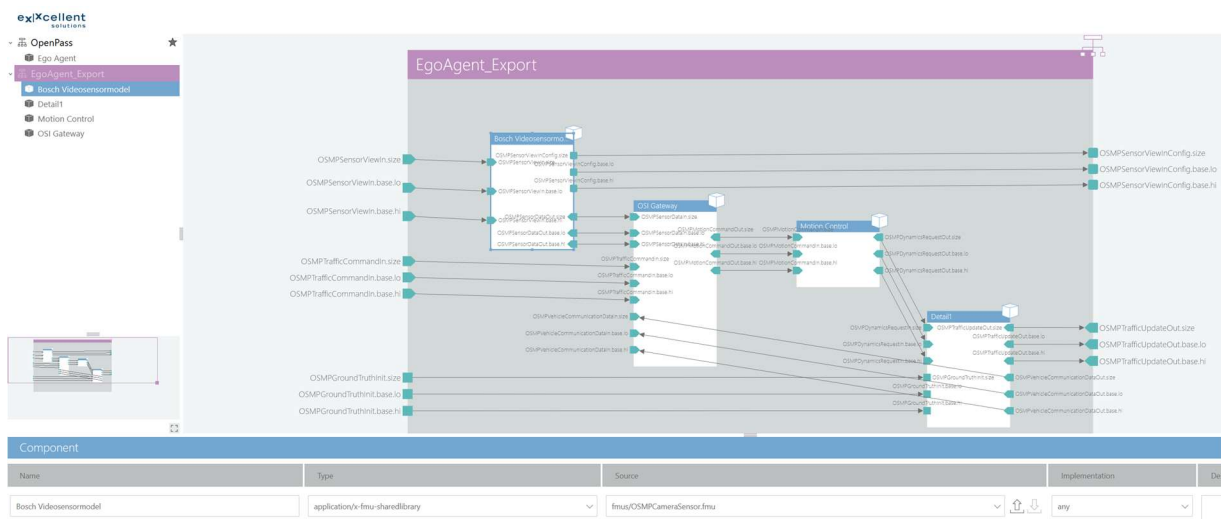


Abbildung 70: SUC1-MS3 Architektur, Sicht aufs Ego Vehicle, in easySSP importiert

In diesem Kontext wurde auch an der Weiterentwicklung und Validierung der SSP-Unterstützung von OpenPASS gearbeitet und zum MS3 ein deutlich reiferer Stand erreicht.

Metadaten

Bedeutung von Metadaten

In Produktentstehungsprozessen ist es regelmäßig *nicht* ausreichend, ein bestimmtes Arbeitsergebnis (Spezifikation, Konstruktion, Prototyp, Validierungsergebnis, usw..) zu erzeugen. Jedes Ergebnis muss immer auch mit Metadaten ausgestattet werden. Selbst wenn nur eine einzelne Person beteiligt wäre, würde sie nach einiger Zeit nicht mehr in der Lage sein, ein bestimmtes Ergebnis zügig aufzufinden oder über dessen Bedeutung und Status Auskunft zu geben. Umso diffiziler ist die Situation, wenn mehrere Personen zusammenarbeiten.

Diese Situation ist selbstverständlich auch im Bereich der Simulation autonomer Fahrfunktionen sehr ausgeprägt. Jedes Objekt (Arbeitsergebnisse, aber auch Simulationswerkzeuge, Prozessschritte usw.) muss mit der richtigen Menge an *Metadaten* ausgestattet sein. Im Rahmen

von SET Level hat AP 4.1 diese Thematik insbesondere im Hinblick auf folgende *Anwendungsgebiete* untersucht:

- Traceability / Verwendungsnachweis für Ergebnisse (sog. „Vorwärtssuche“)
- Verwendete Ressourcen und Eingabedaten („Rückwärtssuche“)
- Überführung der Auswahl in ein ausführbares Modell („Technik“)
- Automatisierbarer Aufbau Gesamtmodell („Delivery“)
- Austausch über Organisationsgrenzen hinweg
- Spezifisch für Verkehrsräumen und Szenarien:
Schnelle Identifikation von zueinander passenden Verkehrsräumen und Szenarien

Besondere Aufmerksamkeit wurde der Identifikation von Metadaten für Verkehrsräume und Szenarien gewidmet.

Arten von Metadaten im Allgemeinen

Als Stand der Technik kann von einer Klassifikation der Metadaten wie im folgenden Kasten ausgegangen werden. SET Level spezifische Umsetzung ist darin ebenfalls mit angegeben:

Descriptive metadata, e.g.:

- model creator's contact address of an artifact
- the context, in which an artifact can be used
- the parameter ranges for which an artifact is validated (if any)
- the model quality of an artifact
- other keywords for information retrieval

Descriptive metadata are typically re-presentable as key-value-pairs.

Structural metadata, e.g.:

- location of an artifact within a hierarchy
 - In SET Level, this is covered by position in GitLab's group/project hierarchy.
- location of a simulation model within a hierarchical simulation model structure
 - In SET Level, this is covered by several means, such as <Catalog/> elements, SSP files, simulator configurations etc.
- "horizontal" relations between artifacts, e.g. SUC → scenario → traffic space → specification
 - In SET Level, for the relation scenario → traffic space, this is covered by position in GitLab's group/project hierarchy. However, for (future) exchange e.g. in OEM-TIER1-situation, an *explicit* representation in SRMD will be mandatory.

Structural metadata are typically re-presentable as relations.

Administrative metadata, e.g.:

- time stamps of creation, last modification
 - In SET Level, this is covered by Git functionality.
- required privileges / information for access control
 - In SET Level, this is covered by GitLab functionality.
- artifact type, e.g. requirement, specification, traffic space, scenario, ...
- release status of an artifact, e.g. draft, ready for test, ready for productive use, withdrawn, ...
 - In SET Level, this is can be achieved using Git tagging functionality.

Administrative metadata usually is mandatory and allows for explicitly enumerated values, only.

Dementsprechend hat AP 4.1 Metadaten für Verkehrsräumen und Szenarien gruppiert.

Kooperation mit TP 3

AP 4.1 hat die Ergebnisse von TP 3 im Hinblick auf Metadaten von Test- bzw. Simulationsfällen, FMU-Modelle und Simulationswerkzeugen reviewed und bei der Formulierung von Use Cases unterstützt.

Metadaten für Verkehrsräumen und Szenarien

Gemeinsam mit TP 1 und TP3 hat AP 4.1 herausgearbeitet, welche Metadaten für Verkehrsräumen und Szenarien relevant sind. Wir sind dabei von den zuvor aufgeführten Anwendungsgebieten ausgegangen und haben die identifizierten Metadaten gemäß der oben gezeigten Darstellung klassifiziert. Die identifizierten Metadaten wurden darüber hinaus danach strukturiert, ob sie

- für Verkehrsräume, (23 Stück)
- für Szenarien, (24 Stück)
- oder für beide Arten von Artefakten (36 Stück)

anwendbar sind.

Die Ergebnisse wurden in https://gitlab.setlevel.de/open/processes_and_traceability/traceability_data/-/tree/main/metadata_for_trafficspaces_and_scenarios veröffentlicht. Die unter <https://gitlab.setlevel.de/open/traffic-spaces-and-scenarios/tss> veröffentlichten Verkehrsräume und Szenarien wurden mit beispielhaften Metadaten in Form von SRMD-Dateien ausgestattet.

Zur Simulation müssen Verkehrsräume und Szenarien die Komplexität und Vielfaltigkeit der realen Welt wiedergeben können. Eine vollständige Verschlagwortung der realen Welt ist aber nicht möglich, schon gar nicht im Rahmen der begrenzten Ressourcen von SET Level. Trotzdem wollen wir Verkehrsräume und Szenarien möglichst prägnant und effizient bedaten. Deshalb wurden folgende Standards und Projektergebnisse Dritter analysiert und in unseren Arbeiten berücksichtigt:

- ASAM OpenLabel
- ASAM OpenODD
- ASAM OpenSCENARIO 1.0
- ASAM OpenSCENARIO 2.0
- ASAM OpenDRIVE
- BSI PAS 1883
- OpenStreetMap
- CityGML

Die eigenen, im Projekt identifizierten Bedarfen an Metadaten wurden mit diesen Quellen abgeglichen und Möglichkeiten zum Re-Use externer Klassifikationsschemata beschrieben. Die meisten dieser Quellen befinden sich gegenwärtig in der Entwicklung – dem Projekt standen aber viele vorläufige Ergebnisse zur Verfügung. Auch aus diesem Grund ist ein flexibler Ansatz notwendig. Entsprechend wurde eine Methode entwickelt, externe Attributierungsschemata mit minimalem Aufwand für SRMD zu erschließen.

SRMD-Datenformat

Als Syntax zur Beschreibung der Metadaten eines Verkehrsraums oder Szenarios wurde das auch an anderen Stellen im Projekt verwendete Format SRMD festgelegt, dann damit wurden insbesondere in TP 3 bereits zuvor im Bereich der Modelle für Fahrzeugkomponenten positive Erfahrungen gesammelt.

Die allgemeine Beschreibung eines Paares aus KEY und VALUE hat demnach in SRMD folgende Gestalt:

```
<?xml version="1.0" encoding="UTF-8"?>
<srmd:SimulationResourceMetaData xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://apps.pmsf.net/STMD/SimulationResourceMetaData SRMD.xsd"
  xmlns:srmd="http://apps.pmsf.net/STMD/SimulationResourceMetaData"
  version="0.5" name="Traffic Space XXX Meta-Data">
  <srmd:Classification type="de.sl4to5.srmd.model-meta-data">
    <srmd:ClassificationEntry keyword="KEY">VALUE</srmd:ClassificationEntry>
  </srmd:Classification>
</srmd:SimulationResourceMetaData>
```

Abbildung 71: Beschreibung eines Paares aus KEY und VALUE in SRMD

In manchen Fällen ist zu beschreiben, dass ein *einzelnes Artefakt* sich tatsächlich aus *mehre- ren Dateien* zusammensetzt. Beispielsweise kann eine Spezifikation bestehen aus

- einer Zielbeschreibung, z. B. als Markdown oder Office-Text,
- einer Lösungsbeschreibung, z. B. als Foliensatz und
- einer Liste von Akzeptanzkriterien, z. B. als CSV-Datei.

Zu diesem Zweck ist es erlaubt, Einträge zum selben KEY, aber mit unterschiedlichem TITLE zu wiederholen:

```
<srmd:ClassificationEntry keyword="model.specification" xlink:href="https://my.server.de/objectives.md" xlink:title="objectives"/>
<srmd:ClassificationEntry keyword="model.specification" xlink:href="https://my.server.de/solution_design.pptx" xlink:title="solution design"/>
<srmd:ClassificationEntry keyword="model.specification" xlink:href="https://my.server.de/acceptance.xlsx" xlink:title="acceptance criteria"/>
```

Abbildung 72: SRMD-Einträge zum selben KEY, aber mit unterschiedlichem TITLE

Soll die Befüllung eines Metadatums nach einem *externen Attributierungsschema* erfolgen, so wurde dafür folgende Syntax vorgeschlagen:

```
<srmd:ClassificationEntry
  keyword="model.classification"
  xlink:href="https://wiki.openstreetmap.org/wiki/Category:Features"
  xlink:title="highway=pedestrian, access:conditional=delivery @ (00:00-11:00), bicycle=yes,
    emergency=fire_hydrant, man_made=obelisk"
/>
```

Abbildung 73: Verwendung von OpenStreetMap als externes Attributierungsschema

Dieses Beispiel folgt dem Attributierungsschema von OpenStreetMap und attribuiert eine Fußgängerzone, in der Lieferverkehr nur bis 11 Uhr erlaubt ist, Fahrräder benutzt werden dürfen und in der es einen Feuerhydranten und einen Obelisken gibt.

Das entwickelte Verfahren eignet sich gleichermaßen, um real *vermessene* Verkehrsräume, daraus abgeleitete *vereinfachte* Verkehrsräume oder völlig synthetisch *erzeugte* Verkehrsräume zu attributieren:

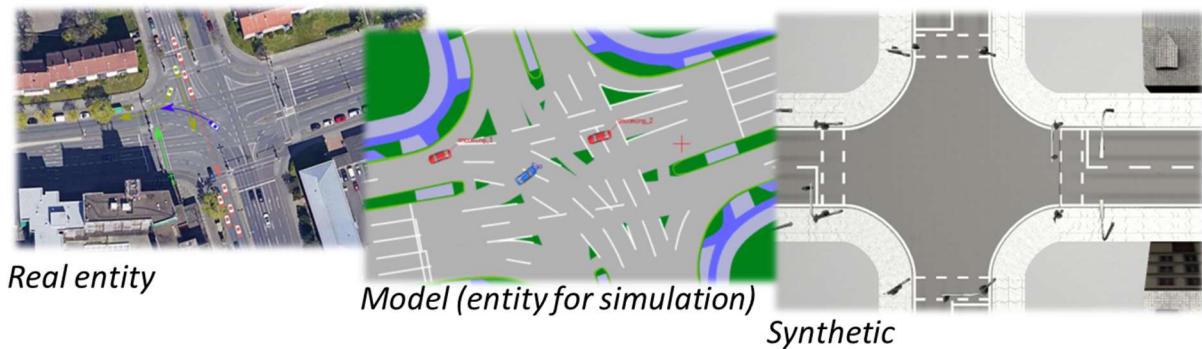


Abbildung 74: Realität – Simulationsmodell – Synthetisches Szenario

Mehr Details zur Syntax wurden in [https://gitlab.setlevel.de/open/processes and traceability/traceability_data/blob/main/SETLevel SRMD and classifications for metadata.pdf](https://gitlab.setlevel.de/open/processes%20and%20traceability/traceability_data/blob/main/SETLevel%20SRMD%20and%20classifications%20for%20metadata.pdf) veröffentlicht.

AP 4.1 hat das erarbeitete Metadatenkonzept mit Beispieldaten aus SUC1 bestätigt. Abbildung 75 zeigt beispielhaft einen Teil der befüllten SRMD-Datei für das SUC1 MS2 Szenario „TS-2-4_RS-2-4“.

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <srmd:SimulationResourceMetaData xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3   xsi:schemaLocation="http://apps.pmsf.net/STMD/SimulationResourceMetaData SRMD.xsd"
4   xmlns:srmd="http://apps.pmsf.net/STMD/SimulationResourceMetaData"
5   version="0.5" name="Scenario Model TS2-4_RS2-4 Meta-Data">
6   <srmd:Classification type="de.sl4to5.srmd.model-meta-data">
7     <!-- Scenario related information -->
8     <srmd:ClassificationEntry keyword="model.type">Scenario</srmd:ClassificationEntry>
9     <srmd:ClassificationEntry keyword="scenario.family">Intersection with occlusion</srmd:ClassificationEntry>
10    <srmd:ClassificationEntry keyword="scenario.maneuver">turn right</srmd:ClassificationEntry>
11    <srmd:ClassificationEntry keyword="scenario.name">TS2-4_RS2-4</srmd:ClassificationEntry>
12    <srmd:ClassificationEntry keyword="scenario.nrtrafficparticip">3</srmd:ClassificationEntry>
13    <srmd:ClassificationEntry keyword="scenario.trafficparticipitypes">Pedestrians (2), vehicle (1)</srmd:ClassificationEntry>
14    <srmd:ClassificationEntry keyword="scenario.environmental">Sunny, 20°C</srmd:ClassificationEntry>
15    <srmd:ClassificationEntry keyword="scenario.handedtraffic">Right-hand traffic</srmd:ClassificationEntry>
16    <srmd:ClassificationEntry keyword="scenario.country-specific.tp">-</srmd:ClassificationEntry>
17    <srmd:ClassificationEntry keyword="scenario.country-specific.sign">-</srmd:ClassificationEntry>
18    <srmd:ClassificationEntry keyword="scenario.sunelevation">90°</srmd:ClassificationEntry>
19    <srmd:ClassificationEntry keyword="scenario.sunazimuth">180°</srmd:ClassificationEntry>
20    <srmd:ClassificationEntry keyword="scenario.wetstreet">>false</srmd:ClassificationEntry>
21
22    <!-- Description of the entity for simulation (aka model) -->
23    <srmd:ClassificationEntry keyword="scenario.provider">SETLevel</srmd:ClassificationEntry>
24    <srmd:ClassificationEntry keyword="scenario.version">1.0.0</srmd:ClassificationEntry>
25    <srmd:ClassificationEntry keyword="scenario.recordingtype">synthetic</srmd:ClassificationEntry>
26    <srmd:ClassificationEntry keyword="scenario.recordingtime">-</srmd:ClassificationEntry>

```

Abbildung 75: Metadaten des Szenarios TS2-4_RS-2-4

Traceability und Traceability-Tool „TRACY“

Traceability

ISO 10007 „Quality management – Guidelines for configuration management“ definiert [12]:

*Configuration management is a **management activity** that applies technical and administrative direction over the life cycle of a product and service, its configuration **identification** and **status**, and related product and service configuration **information**.*

Wikipedia führt dazu aus: [16]

Its guidance is specifically recommended for meeting “the product identification and traceability requirements” introduced in ISO 9001:2015 “Quality management systems – Requirements” and AS9100 Rev D “Quality Management Systems – Requirements for Aviation, Space, and Defense Organizations”.

Im Kontext des hoch automatisierten Fahrens treffen wir auf ähnlich komplexe Systeme und ähnlich hohe Ansprüche an Qualität und Zuverlässigkeit wie im Bereich der Luftfahrt. Daher ist es folgerichtig, sich an den entsprechenden Normen und Vorgehensweisen zu orientieren.

Zum Nachweis der Sicherheit und anderer Qualitäten eines hoch automatisierten Fahrzeugs ist es unabdingbar, seine exakte Konfiguration dokumentiert zu haben, die Zusammenstellung und Konfiguration der zugehörigen Entwicklungsartefakte zu managen (ISO 10007 [12]) und jederzeit den Nachweis führen zu können, wie jedes einzelne Artefakt zu Stande gekommen ist. Letzteres versteht man unter dem Begriff „Traceability“. Diese Traceability ist auch essenzielle Grundlage für die Glaubwürdigkeit von Entwicklungsergebnissen. TP 3 befasste sich unter dem Oberbegriff „Credible Simulation Process (CSP)“ intensiv damit.

ISO 10007

- 1 Scope
- 2 Normative references
- 3 Terms and definitions
- 4 Configuration management responsibility
 - 4.1 Responsibilities and authorities
 - 4.2 Dispositioning authority
- 5 Configuration management process
 - 5.1 General
 - 5.2 Configuration **management** planning
 - 5.3 Configuration **identification**
 - 5.3.1 Product structure or service capability and selection of configuration items
 - 5.3.2 Configuration information
 - 5.3.3 Configuration baselines
 - 5.4 Change **control**
 - 5.4.1 General
 - 5.4.2 Initiation, identification and documentation of the need for change
 - 5.4.3 Evaluation of change
 - 5.4.4 Disposition of change
 - 5.4.5 Implementation and verification of change
 - 5.5 Configuration status **accounting**
 - 5.5.1 General
 - 5.5.2 Documented Information
 - 5.5.3 Reports
 - 5.6 Configuration **audit**

Abbildung 76: Inhaltsverzeichnis der ISO 10007 [12]

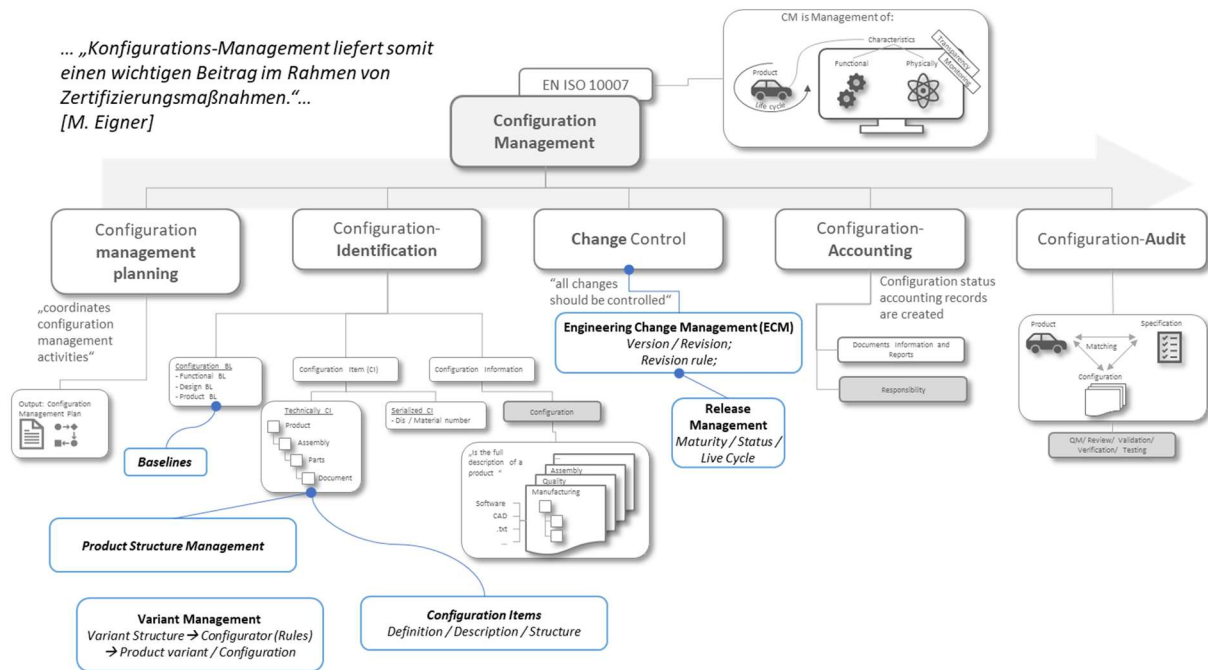


Abbildung 77: Konfigurationsmanagementbereiche gemäß ISO 10007

TRACY – das Traceability-Tool

Ohne die Unterstützung eines leistungsfähigen Software-Werkzeugs ist es praktisch unmöglich, die Vielzahl an Artefakten, Teilprozessen und ihrer wechselseitigen Abhängigkeiten unter Kontrolle zu halten.

Daher wurde im Rahmen von SET Level ein solches Software-Werkzeug als Demonstrator für Traceability unter dem Namen „TRACY“ entwickelt. Zu seiner Validierung wurde der CSP verwendet.

TRACY – Use Cases und Anforderungen

Bereits zu Projektbeginn wurde z. B. folgender Use Cases identifiziert:

- TRACY muss dokumentieren können:
 - Welche Szenarien wurden umgesetzt?
 - Welche Modelle waren dazu nötig?
 - Wie waren die Simulationstools konfiguriert?
 - Welche Ergebnisse haben die Simulationsläufe gezeigt?

Die an AP 4.1 beteiligten Projektpartner haben sich an der Anforderungsspezifikation für TRACY aus den SUCs heraus beteiligt. Auf Grund der Bedeutung des CSPs fand eine besonders enge Abstimmung mit TP 3 statt.

TRACY – Artefakt-Arten

TRACY unterstützt u. a. folgende *Artefakt-Arten*:

- Requirement
- User Story
- Specification
- Test Case
- Model
- System / Systembestandteil

Für jedes Artefakt dieser Artefakt-Arten stehen folgende *Metadaten* zur Verfügung:

- id
- title
- rationale
- linkToResource
- repository
- idName
- displayName
- resourceType
- uri

Auch *eigene Metadaten* können angehängt werden. Alle Objekte verfügen dazu über eine Liste *generischer Key/Value/Type-Attribute*. Zulässige Typen sind Text, Numerisch, URL, Datum, Zeitstempel.

Darüber hinaus können Objekte mit *Tags* versehen werden.

Alle Informationen werden entsprechend des im TP 3 entwickelten **Credible Simulation Process (CSP)** aufgeführt und können insbesondere auf ihre Relationen hin untersucht werden (z. B. welches Modell welche Anforderung erfüllen muss).

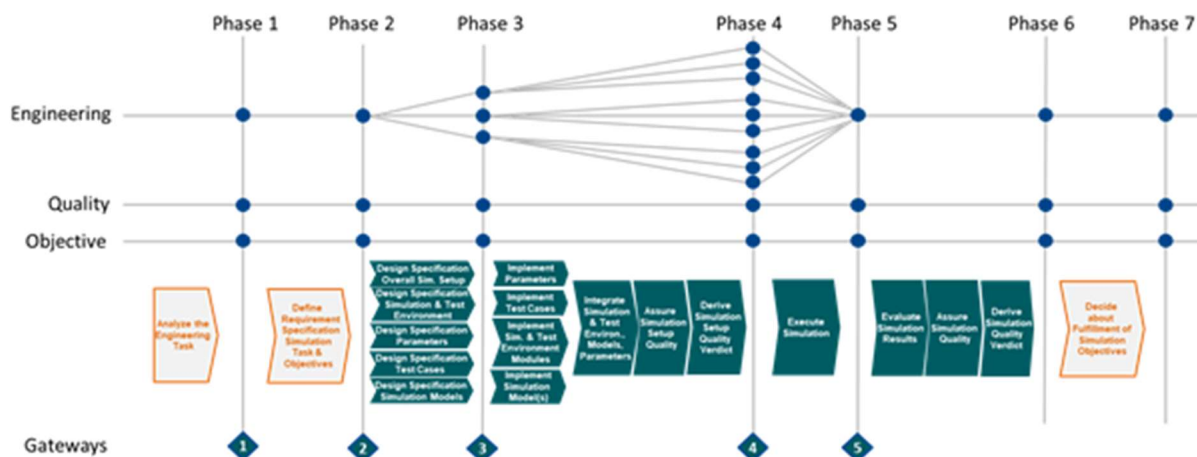


Abbildung 78: Schematische Darstellung der Artefakt-Relationen über den CSP hinweg

Der CSP wird von TRACY vollständig unterstützt.

TRACY – Funktionalitäten

Um die Anforderungen abzudecken, wurden in TRACY u. a. folgende Funktionalitäten realisiert:

- User-Management
- Rollen- und Rechte-Management
- Grafische Benutzeroberfläche (GUI)
- User-spezifische Cockpits und Sichten
- Anlegen und Suchen von Artefakten (inkl. Metadaten)
- Anlegen von Trace-Links zwischen Artefakten (Relationen)
- Abbilden des Credible Simulation Processes

- Befüllen von TRACY mit Projektdaten
- Workflow zur Qualitätssicherung entsprechend CSP
- Einbindung weiterer Systeme nach dem Standard OSLC (beispielsweise DOORS)
- Grafische Darstellung von verwalteten Artefakten und ihrer Relationen
- Möglichkeiten zur Durchführung dynamischer Impact-Analysen (Filter)
- Realisierung einer ersten Baseline-Funktionalität
- Aktualisierung der CSP- und Einführung der CMP-Templates
- Erweiterung der Darstellungs- und Filtermechanismen in der Graph-Ansicht
- Realisierung von Notifications (Hinweise, z. B. wenn einem Benutzer ein neuer Task zugewiesen wurde)
- Cloudfähigkeit der TRACY-Software
- Löschen von den in TRACY verwendeten Entitäten
- Abspeichern von ersten individuellen User Settings

Auf einige dieser Funktionalitäten gehen wir nun näher ein.

TRACY – Benutzeroberfläche

TRACYS grafische Benutzeroberfläche (GUI) ist in drei Säulen gegliedert:

Die Graph-Darstellung der Informationsstrukturen steht im Zentrum der Applikationsoberfläche. Hier hat der Benutzer die Möglichkeit, über Filter Impact-Analysen durchzuführen. Die dargestellten Informationen wurden zudem angereichert. Beispielsweise wird der Reifegrad der Informationen angezeigt und auch dargestellt, ob es sich um verlinkte oder ausschließlich in TRACY gehaltene Informationen handelt.

Auf der linken Seite findet der Anwender die globale Navigationsleiste und daneben die Strukturierung der Informationen.

Rechts vom Graph findet der Nutzer Details zu selektierten Informationen, Metainformationen, Status, Historie und Abhängigkeiten:

The screenshot displays the TRACY web application interface. At the top, there is a navigation bar with tabs for 'DASHBOARD', 'PROJECTS', 'TASKS', and 'ARTIFACTS'. The main area is divided into three columns:

- Left Column (Navigation):** A sidebar with a 'Dashboard' section containing 'Navigator', 'Impact Analysis', and 'Baseline Report'. Below this is a 'Global' section with 'Search', 'Link Wizard', 'Project List', 'Profile', and 'About'.
- Middle Column (Graph View):** A central workspace showing a hierarchical tree view on the left and a graph view on the right. The tree view shows a project structure under 'SUC1 Criticality Analysis MS2', including 'Credible Simulation Process', 'Analysis Phase SUC1 MS2', and various sub-phases. The graph view shows a network of nodes and relationships, with a search filter set to 'Artifacts, Projects, Activities (Phases/P...)' and a 'Level of depth' set to 3.
- Right Column (Entity Data):** A detailed view of a selected entity, 'SUC1 Criticality Analysis MS2'. It includes fields for 'Project ID' (PR-CRIT-MS2), 'Manager' (Jane P. Miller), 'Start and Target Date' (01/03/2020 - 30/06/2021), and 'Description'. Below this are sections for 'Targets', 'Metadata', 'History' (listing creation and editing dates and users), 'Relations (1)', 'Assignments (0)', and 'Trace Links (0)'.

Abbildung 79: Gliederung der TRACY-GUI

TRACY – Metadaten-Funktionalitäten

Für das Konsortium ist es von entscheidender Bedeutung, neben den eigentlichen Modelldaten auch deren **Metadaten** zentral und einfach pflegen, suchen etc. zu können. Dazu wurde in TRACY ein Editor realisiert, der es jedem Anwender ermöglicht, Metadaten zu editieren und abzuspeichern. So ermöglicht TRACY dem Benutzer ein einfaches Management der jeweiligen Artefakte, das Pflegen der zugehörigen Metadaten sowie weiterer Daten wie Statusinformation, Zuständigkeiten, Anmerkungen (Rationales) etc. in einer Oberfläche.

Dabei verwaltet TRACY die Metadaten der Artefakte sowie deren Beziehungen untereinander, nicht aber die Artefakte selbst, die im GitLab persistiert sind.

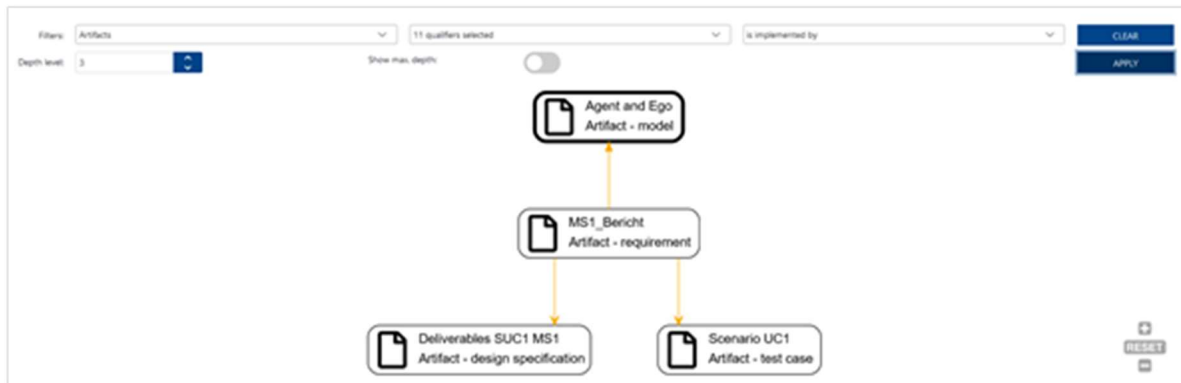


Abbildung 80: Graphische Darstellung von Artefakten und ihren Relationen (Filter aktiviert)

TRACY – Impact-Analyse

Relevante Informationen in Bezug zu setzen ist eine der wesentlichen Funktionalitäten von TRACY. Um effizient damit arbeiten zu können, benötigen Anwender jedoch auch visuelle bzw. graphische Unterstützung und die Möglichkeiten, nach gewissen Informationen zu suchen und zu filtern. Hierunter fallen sogenannte **Impact-Analysen**, die Auskunft darüber geben, welche Artefakte Einfluss auf welche anderen Artefakte haben. Anwender möchten zum Beispiel danach filtern können, welche Requirements mittelbaren oder unmittelbaren Einfluss auf die Modellentwicklung haben.

Durch einfaches Setzen von Filtern kann ein Anwender dies in TRACY grafisch visualisieren. In Abbildung 81 ist exemplarisch gezeigt, welche Projekte von einer spezifischen Anforderung (Requirement) beeinflusst werden. Ein Anwender kann sich so schnell einen Überblick darüber verschaffen, was z. B. bei einer Änderung der Anforderung betroffen ist und wer zu informieren

ist. Generell können diese Abhängigkeitsuntersuchungen zwischen jeder Art von Artefakten durchgeführt werden.

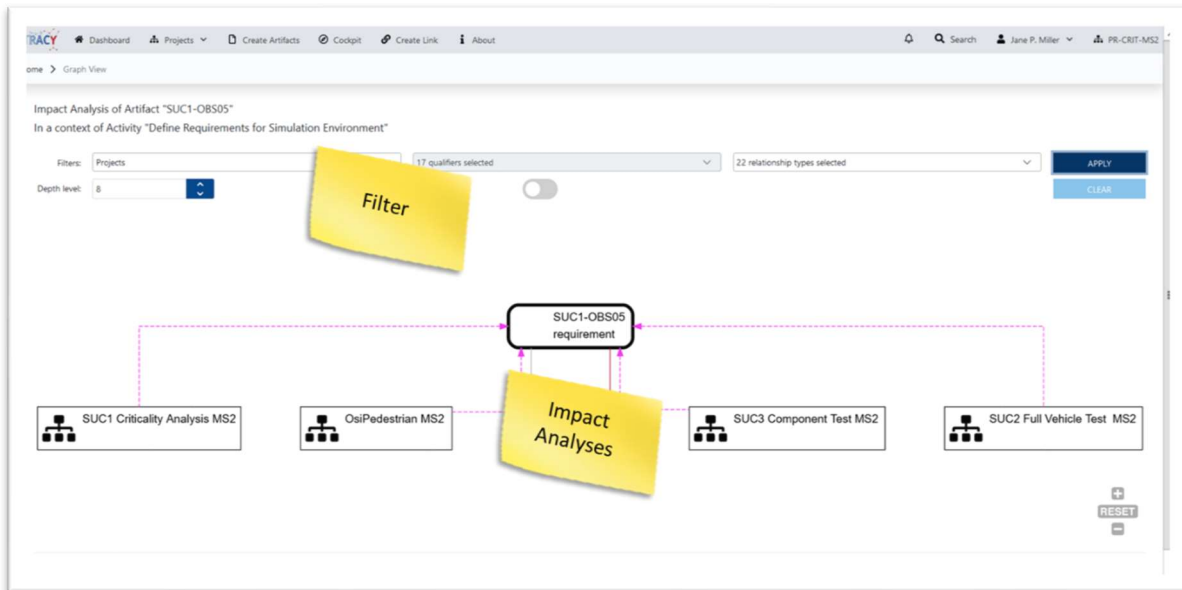


Abbildung 81: Exemplarische Anwendung des grafischen Filtermechanismus

TRACY – Grafische Analyse

Ein weiteres Feature ist die Graphenanalyse, die die Impact-Analyse ergänzt und unterstützt. Neben der generischen Analyse, die durch komplexe Benutzereingaben konfigurierbar ist, wurden vorkonfigurierte Analysen implementiert.

In der *Where-Used-Analyse* werden alle Projekte gelistet, in denen die betrachtete Entität genutzt wird. Für Configuration Items sind zudem noch eine *Show-Impacted-Entity*- und eine *Show-Impacting-Entity*-Funktionalität zur Verfügung gestellt worden.

Diese Analysen ziehen automatisch Verbindungen zwischen Configuration Items, die sich gegenseitig beeinflussen. Der Einfluss wird dabei durch die Verbindung über Planungselemente, wie bspw. Aktivitäten, und deren Relationen dazu (Input, Guideline und Output, Rationale) ermittelt.

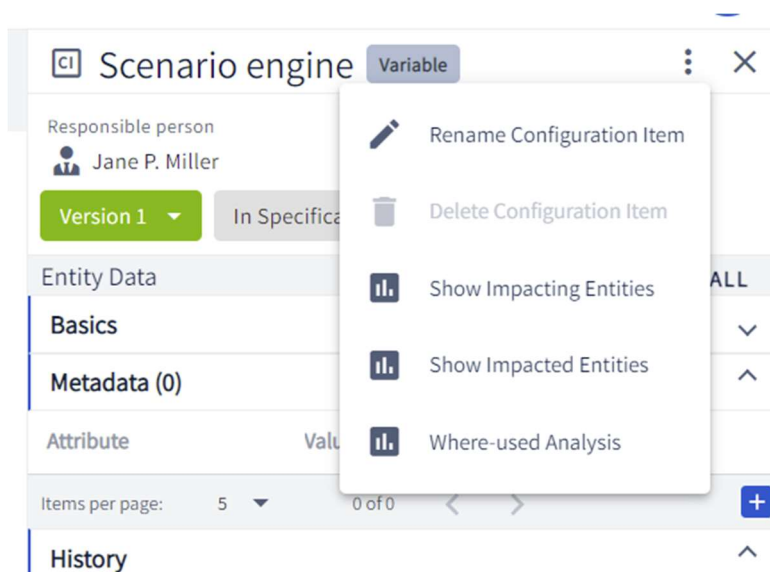


Abbildung 82: Auswahl der möglichen vorkonfigurierten Analysen

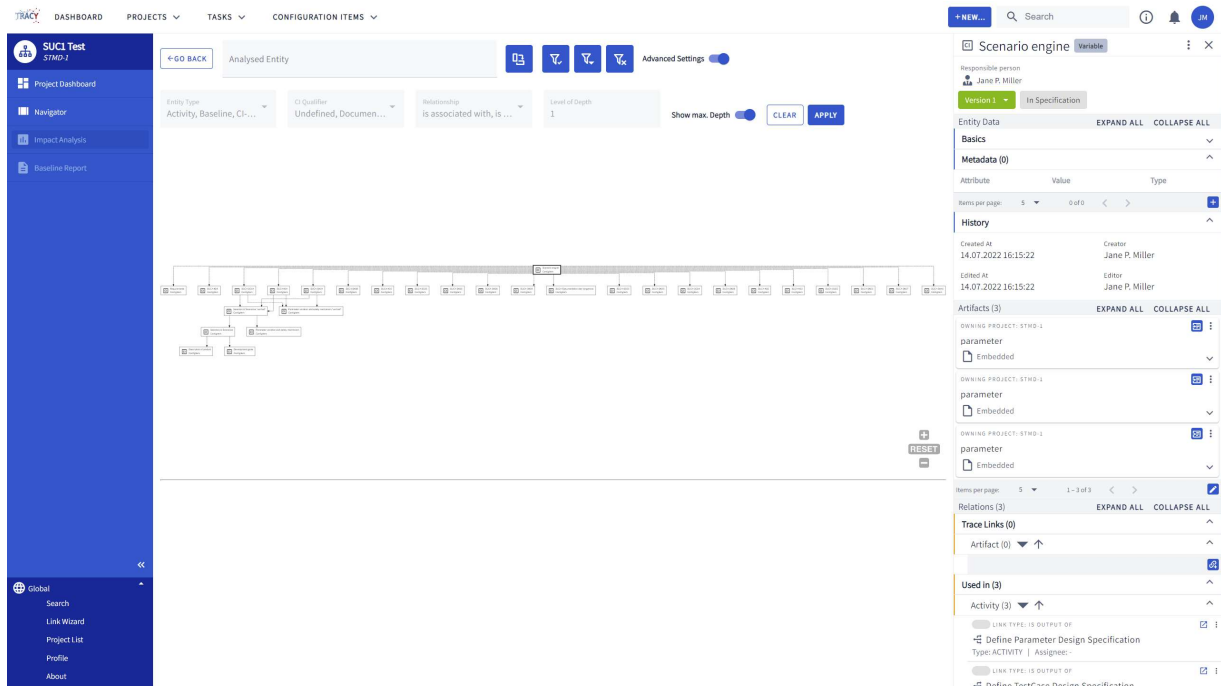


Abbildung 83: Exemplarische Durchführung der Impacted-Entities-Analyse
TRACY – Baselining

Mit einer ersten **Baseline-Lösung** (sog. *Ad-Hoc Baselines*) konnten selektierte Artefakte gesichert und mit einem Zeitstempel versehen werden.

Nachfolgend wurden die Erstellung von Baselines und das Ableiten von Reports weiterentwickelt. Baselines dienen dazu, dass Arbeitsstände zu einem beliebigen Zeitpunkt t „eingefroren“ werden und somit jederzeit nachvollziehbar ist, was der Stand der Arbeit zum Zeitpunkt t war (z. B. der Stand zu einem Berichtszeitpunkt oder der Stand der Ergebnisse, die mit einem Partner ausgetauscht wurden, oder der Stand zum Projektende etc.). Darauf aufbauend wurde eine Report-Funktion in TRACY realisiert, um einen Übersichtsbericht basierend auf der Baseline abzuleiten.

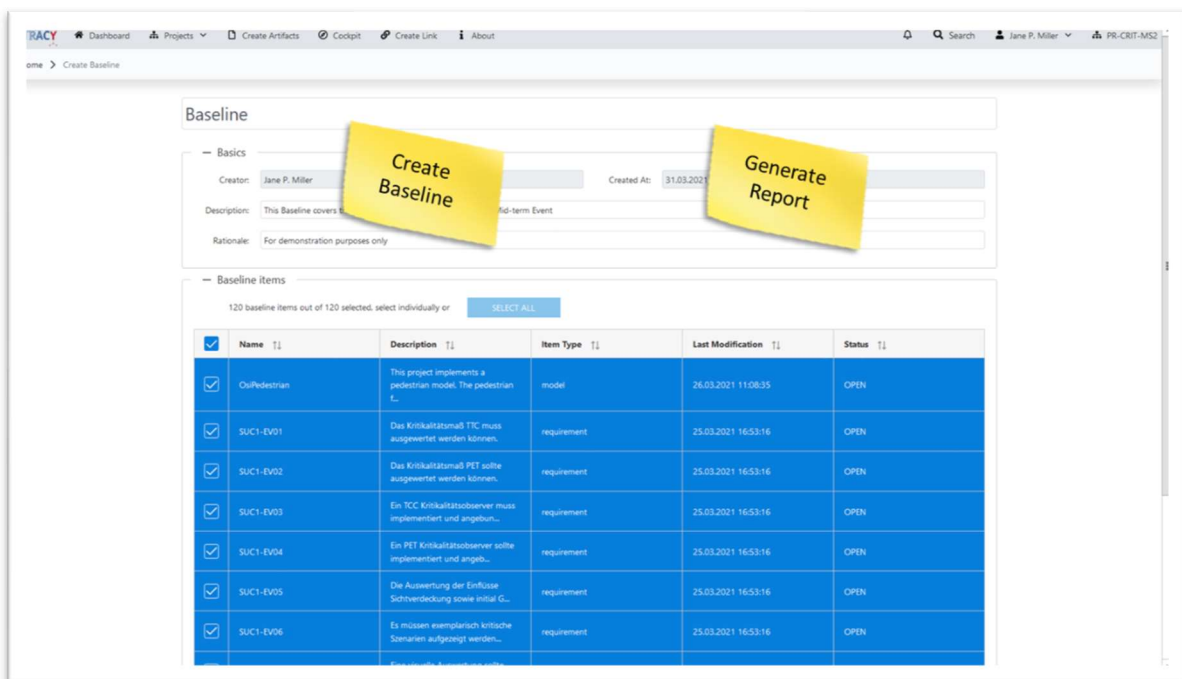


Abbildung 84: Exemplarische Darstellung zum Ableiten einer Baseline

Planned Baselines sind Vorlagen (Templates) für die oben beschriebenen Ad-Hoc Baselines. Eine Planned Baseline definiert Traversierungsregeln, um durch den Graphen der Datenbasis zu navigieren und außerdem Auswahlfilter dafür, welche Entitäten in einer gezogenen Ad-Hoc Baseline landen sollen. Ergänzt durch einen *Trigger*, der aus einem Element (bspw. Aktivität) und einem Trigger-Status (Release Status) besteht, kann die Planned Baseline automatisch instanziiert werden, sobald das Trigger-Element den angegebenen Trigger-Status erreicht. In dem Fall wird aus der Vorlage eine Ad-Hoc Baseline erzeugt, um die durch die Traversierungs- und Filterregeln definierte Datenmenge einzufrieren. Abgesehen von dem automatischen Trigger ist auch ein manuelles Auslösen einer Planned Baseline möglich.

Abbildung 85: Planned Baseline im Navigator (Tree View links) und Detail View (rechts) mit ausgeführter Ad-Hoc Baseline

TRACYs Datenmodell gruppiert Objektrelationen und Trace-Links in sog. „Link Familien“, die bei Abfragen und Analysen einheitlich verwendet werden können. Aktivitäten wurden verallgemeinert, sodass Prozesse, Phasen und Aktivitäten lediglich durch einen Qualifier spezifiziert werden und keine unterschiedlichen Klassen sind.

TRACY – Configuration Items

Configuration Items (CIs) stellen eine Art Hülle um Artefakte dar, die eine Versionierung zulassen. Meta-Informationen, wie Qualifier, Release Status, Beschreibung, Bearbeiter, etc. werden dem CI zugeordnet. So kann sichergestellt werden, dass Artefakte stets unveränderlich bleiben und die Evolution eines Configuration Items über den Simulationsprozess dargestellt werden kann.

Beispiel: Die Anforderung an ein Modell beginnt als Einzeiler (Version 1), wird anschließend in einer Aktivität ausgearbeitet und ergibt ein Dokument (Version 2). Im Laufe der Simulationsiterationen werden die Anforderungen ggf. überarbeitet (Version 3 und ggf. weitere Versionen). Am Ende der Simulation steht die aktuelle Version der Anforderung, aber in der Datenmenge ist erkenntlich, zu welchem Arbeitsschritt welches Artefakt für diese Anforderung vorgelegen hat. Entsprechend wurden Trace Links und Relationen in TRACY so erweitert, dass entweder eine konkrete CI-Version oder eine Laderegel (bspw. aktuelle Version) hinterlegt werden kann.

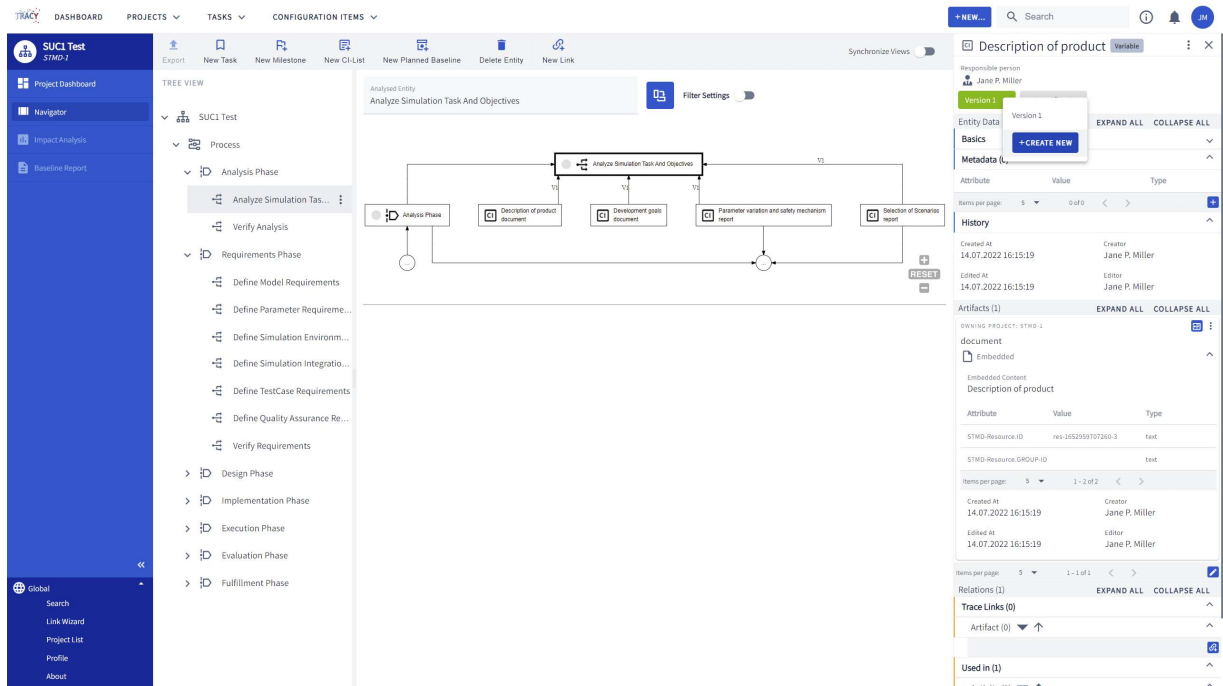


Abbildung 86: Ansicht eines Configuration Items in der Detail View (rechts) mit Auswahlmöglichkeit der Version

Configuration Items können zusätzlich genutzt werden, um mehrere Artefakte zu einem Objekt zusammenzufassen (sog. *Composite* Configuration Item). Dies ist beispielsweise von Nutzen, wenn mehrere Dokumente ein Modell beschreiben.

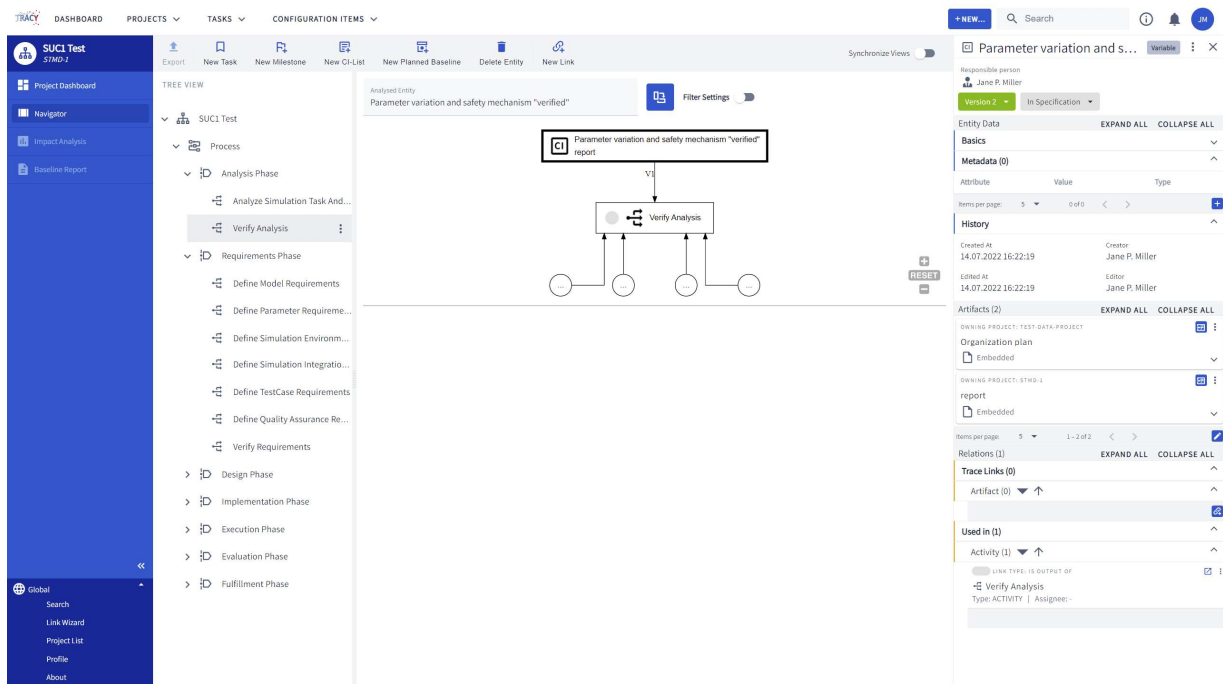


Abbildung 87: Darstellung eines Composite Configuration Items in der Detail View (rechts)

TRACY – Projektverwaltung

TRACY bietet eine **Projektverwaltung**. Im Rahmen solcher Projekte sind Personen Rollen zugewiesen. Prozesse und Artefakte haben einen Bezug zu Projekten.

Ein Projekt kann geschlossen werden und wird dadurch eingefroren. Geschlossene Projekte können aber auch wieder geöffnet oder gelöscht werden. Zudem wurde ein Projektexport in

einem TRACY-nativen XML-Format implementiert, mit dem Daten ohne Verlust zwischen TRACY Instanzen ausgetauscht werden können.

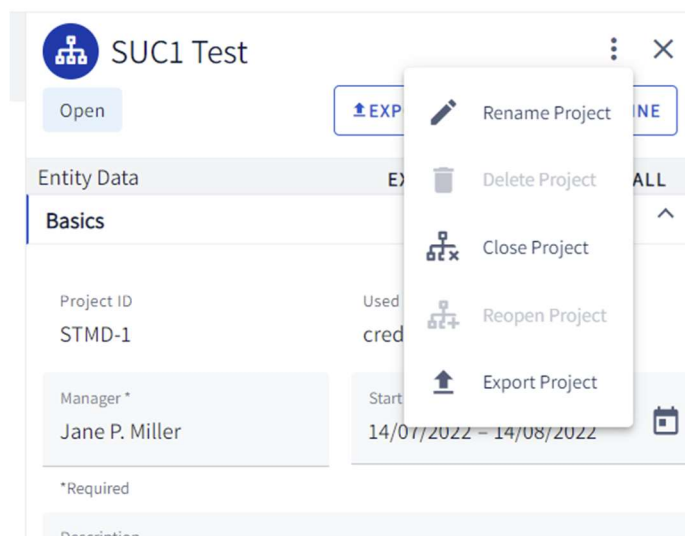


Abbildung 88: Löschen, Schließen, Öffnen und Exportieren von Projekten

TRACY – Dashboard

Zusätzlich zum globalen **Dashboard** (Überblick über *mehrere* Projekte) wurde ein projektspezifisches Dashboard eingeführt. Dieses Projekt-Dashboard zeigt lediglich Elemente an, die im Kontext *eines* Projektes angelegt bzw. genutzt wurden. Neben den bekannten Kacheln für Aktivitäten und Configuration Items wurden Kacheln für zugewiesene Elemente in bestimmten Release Phasen („In Work“ und „In Review“), letzte Aktivitäten im Projekt, sowie ein Gantt Chart zur Projektübersicht hinzugefügt. Die Kachel für Projekte ist im Projekt Dashboard selbst nicht vorhanden.

TRACY – Rollen & Rechte

TRACYs Rollen- und Rechte-Management umfasst u. a. einen Release-Prozess mit Zugriffsrechten derart, dass nur die aktuell zugewiesene Person die Aktivitäten bearbeiten kann. Die Fortschreitung des Release-Prozesses bleibt dabei unverändert. Auf Configuration Items wurde derselbe Release Prozess inkl. Zugriffsrechten angewandt.

TRACY – Schnittstellen

Multi-System-Connectivity ist für Traceability allgemein von entscheidender Bedeutung, da die Informationen von Interesse oft über eine Vielzahl von Systemen hinweg verstreut sind und erst über TRACY in Bezug zueinander gesetzt werden können.

Für SET Level ist es von besonderer Bedeutung, dass TRACY auf *Artefakte im GitLab* zugreifen und sie persistent verlinken kann. TRACY besitzt daher eine **Git-Schnittstelle**, so dass es sich mit dem projekteigenen GitLab-Server verbinden kann.

Der standard-basierte OSLC-Connector ermöglicht das Einbinden einer Vielzahl von Systemen.

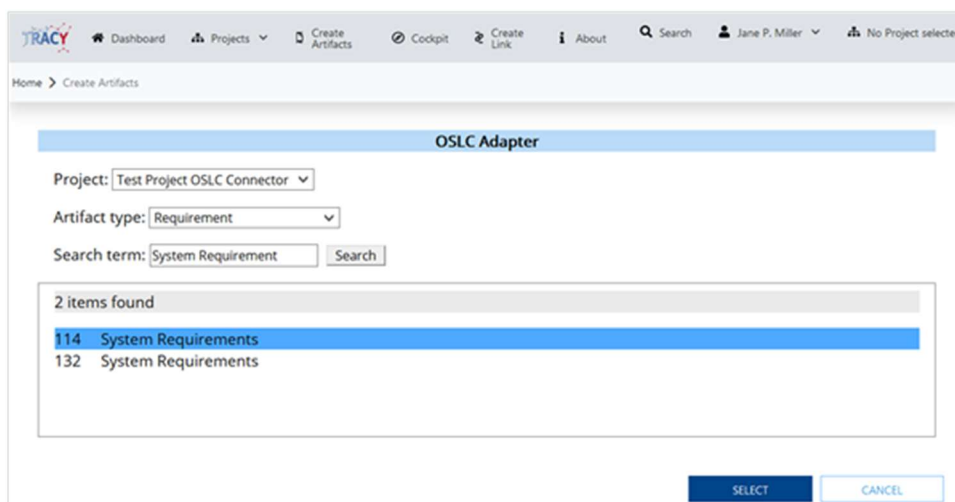


Abbildung 89: TRACY OSLC Connector, exemplarisches Einbinden von Requirements

TRACY – Erprobung

Während der gesamten Laufzeit von SET Level wurden neu entwickelte TRACY-Funktionalitäten umgehend dem Projekt zur Verfügung gestellt und erprobt. Die im Kontext der SUCs erzeugten Artefakte samt der Verkehrsräume und Szenarien dienen dabei direkt als praxisnahe Testdaten.

Das Befüllen von TRACY mit diesen Testdaten erfolgte in enger Zusammenarbeit mit den SUCs und AP 2.1.

Auf Betreiben der Industriepartner wurde TRACY in der Cloud bereitgestellt, um so eine schnellere und individuelle Verprobung auch mit eigenen Daten realisieren zu können. Zuvor stand dem Projekt lediglich eine einzelne, zentrale TRACY-Instanz zur Verfügung, in der die Arbeiten der SUCs dokumentiert und die im GitLab zur Verfügung gestellten Informationen und Modelle verlinkt wurden.

So konnten die Industrialisierbarkeit der Projektergebnisse in den Häusern der Partner vorangetrieben werden.

Der Einsatz von Cloudinstanzen wurde von den Projektpartnern als sehr wertvoll angesehen, war aber in der ursprünglichen VHB nicht geplant. Im Einzelnen wurden Instanzen für VW, BMW, Bosch, Continental, Ford und Opel bereitgestellt.

Die gesammelten Erfahrungen wurden über TP 5 in die anderen Teilprojekte zurückgespiegelt.

TRACYs Issue Management Process

Wie oben beschrieben, wurde TRACY in enger Folge der Partnerwünsche dynamisch weiterentwickelt. Um das Vorgehen dabei zu veranschaulichen, wurde ein prototypischer Prozess für das Issue- und Anforderungsmanagement für ein Simulationswerkzeug anhand des Beispiels TRACY dargestellt:

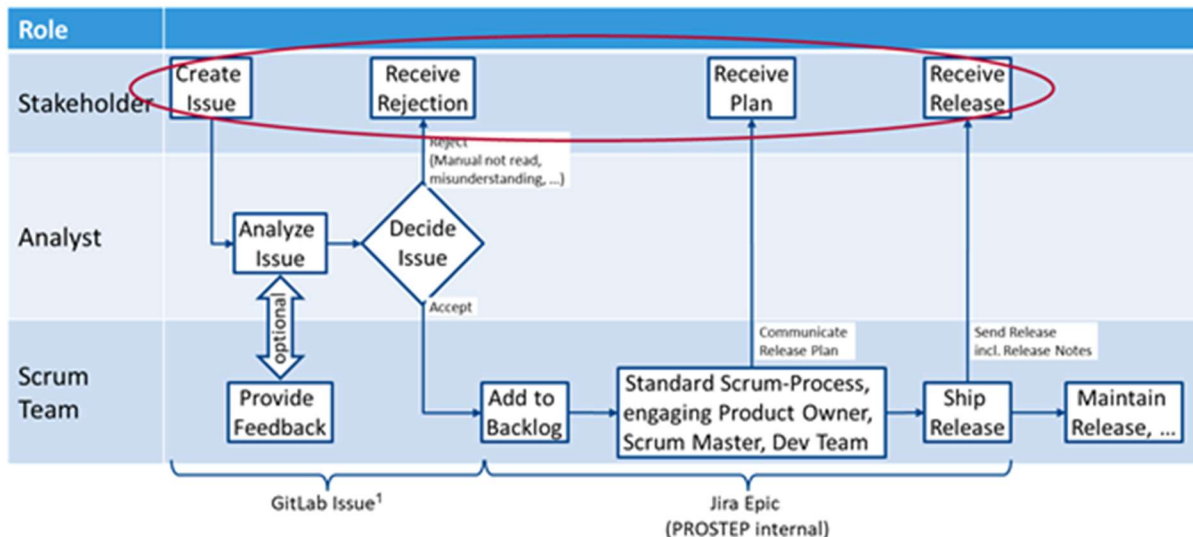


Abbildung 90: TRACY Issue Management Prozess

TRACY – Projekt-externe Darstellung

TRACY wurde u. a. auf folgenden Veranstaltungen mit Vorträgen präsentiert:

- prostep ivip Symposium 2020
- 6th Symposium for Driving Simulation (SDS) 2020
- Modelica Conference 2021
- Workshop der prostep ivip Projektgruppe SmartSE 2021
- öffentliches Webinar des prostep ivip Vereins 2021
- prostep ivip Symposium 2022

Zu allen Vorträgen wurde auch intensiv in Social Media wie LinkedIn und YouTube berichtet.

1.1.4.2 AP 4.2: Kritikalitätsanalyse

Obwohl von der VHB ursprünglich nicht vorgesehen, hat sich PROSTEP auch in AP 4.2 engagiert. Dies war u. a. erforderlich, um eine wirklich SUC-übergreifende integrative Architektur der Simulation erarbeiten zu können.

Stichpunkthaft umfasste das Engagement insbesondere:

- Beteiligung in den regelmäßigen Meetings
- Unterstützung bzgl. der Umsetzung und Erprobung des CSP
- Anleitung bzgl. des Versionsmanagements der erzeugten Artefakte mittels GitLab
- Unterstützung bei der Definition des Change-Management-Prozesses für die erzeugten Artefakte
- Unterstützung bei der Ausstattung der erzeugten Artefakte mit Lizenzen
- Analyse der Simulationsziele des SUC. Schlussfolgerungen daraus für die integrative Simulationsarchitektur
- Modellierung der integrativen Simulationsarchitektur in enger Abstimmung mit dem SUC
- Ausstattung der erzeugten Artefakte mit Metadaten in enger Abstimmung mit dem SUC
- Definition der Metadaten, die mittels des SRMD-Formats serialisiert werden sollen
- Exemplarische Bewertung der Laufzeit-Performance erstellter Verhaltensmodelle
- Darstellung der Traceability der vom SUC verwendeten und erzeugten Artefakte mittels TRACY

Die Zusammenarbeit wurde von AP 4.1 aus gesteuert. Viele Details dazu finden sich dementsprechend im obigen Kapitel AP4.1. Details zur Arbeit am CSP sind im Abschnitt zu TP 3

enthalten. Über Arbeiten im Kontext der Erfassung und Analyse von Anforderungen wird im Abschnitt zu TP 1 berichtet, da sie gemeinsam mit TP 1 und VV-Methoden erfolgten.

1.1.4.3 AP 4.3 Level 4 und 5 Systeme

Stichpunkthaft umfasste das Engagement der PROSTEP in AP 4.3 insbesondere:

- Beteiligung in den regelmäßigen Meetings
- Unterstützung bzgl. der Umsetzung und Erprobung des CSP
- Anleitung bzgl. des Versionsmanagements der erzeugten Artefakte mittels GitLab
- Unterstützung bei der Definition des Change-Management-Prozesses für die erzeugten Artefakte
- Unterstützung bei der Ausstattung der erzeugten Artefakte mit Lizenzen (Wie oben erwähnt, wurde vom Projekt die Lizenz „Creative Commons Attribution-ShareAlike 4.0 (CC BY-SA 4.0)“ als Default ausgewählt.)
- Analyse der Simulationsziele des SUC. Schlussfolgerungen daraus für die integrative Simulationsarchitektur
- Modellierung der integrativen Simulationsarchitektur in enger Abstimmung mit dem SUC
- Ausstattung der erzeugten Artefakte mit Metadaten in enger Abstimmung mit dem SUC
- Definition der Metadaten, die mittels des SRMD-Formats serialisiert werden sollen
- Exemplarische Bewertung der Laufzeit-Performance erstellter Verhaltensmodelle
- Darstellung der Traceability der vom SUC verwendeten und erzeugten Artefakte mittels TRACY

Die Zusammenarbeit wurde von AP 4.1 aus gesteuert. Viele Details dazu finden sich dementsprechend im obigen Kapitel zu AP4.1. Details zur Arbeit am CSP sind im Abschnitt zu TP 3 enthalten. Über Arbeiten im Kontext der Erfassung und Analyse von Anforderungen wird im Abschnitt zu TP 1 berichtet, da sie gemeinsam mit TP 1 und VV-Methoden erfolgten.

1.1.4.4 AP 4.4 Komponentensimulation

Stichpunkthaft umfasste das Engagement der PROSTEP in AP 4.4 insbesondere:

- Beteiligung in den regelmäßigen Meetings
- Unterstützung bzgl. der Umsetzung und Erprobung des CSP
- Anleitung bzgl. des Versionsmanagements der erzeugten Artefakte mittels GitLab
- Unterstützung bei der Definition des Change-Management-Prozesses für die erzeugten Artefakte
- Unterstützung bei der Ausstattung der erzeugten Artefakte mit Lizenzen (Wie oben erwähnt, wurde vom Projekt die Lizenz „Creative Commons Attribution-ShareAlike 4.0 (CC BY-SA 4.0)“ als Default ausgewählt.)
- Analyse der Simulationsziele des SUC. Schlussfolgerungen daraus für die integrative Simulationsarchitektur
- Modellierung der integrativen Simulationsarchitektur in enger Abstimmung mit dem SUC
- Ausstattung der erzeugten Artefakte mit Metadaten in enger Abstimmung mit dem SUC
- Definition der Metadaten, die mittels des SRMD-Formats serialisiert werden sollen
- Exemplarische Bewertung der Laufzeit-Performance erstellter Verhaltensmodelle
- Darstellung der Traceability der vom SUC verwendeten und erzeugten Artefakte mittels TRACY

Die Zusammenarbeit wurde von AP 4.1 aus gesteuert. Viele Details dazu wurden oben unter AP4.1 beschrieben. Details zur Arbeit am CSP sind im Abschnitt zu TP 3 enthalten. Über Arbeiten im Kontext der Erfassung und Analyse von Anforderungen wird im Abschnitt zu TP 1 berichtet, da sie gemeinsam mit TP 1 und VV-Methoden erfolgten.

1.1.5 Teilprojekt 5: Einbettung und kritische Reflexion

1.1.5.1 AP 5.1 Proof of Concept

Im Rahmen des SET Level Projektes wurde im „Proof of Concept“ ein Referenzprozess dokumentiert. Dieser soll die Prüfung auf konzeptioneller Ebene und den dazugehörigen Review unterstützen. Der Prozess dient dabei, dazu ein einheitliches Vorgehen der Projektpartner bei Teilen der Durchführung des „Proof of Concept“ zu gewährleisten. Hierbei wird der in AP5.2 entwickelte Einführungsprozess zur industriellen Erprobung als Basis genutzt.

Zusätzlich wurde das der „Prozesssicheren Gesamtbeschreibung“ zugrunde liegende Vorgehen, welches im AP5.1 finalisiert wurde, ebenfalls als Prozess dokumentiert und in die Gesamtdarstellung integriert. Die Gesamtdarstellung beinhaltet damit zwei verschiedene Stränge. Einerseits beschreibt dabei der erste Strang die Prüfung einzelner ausgewählter Projektergebnisse auf Basis eines Prüfkriterienkatalogs in Form von Reviews mit den Ergebnisverantwortlichen. Andererseits beschreibt der zweite Strang das Vorgehen zur Prüfung von Projektergebnissen in deren Zusammenwirken anhand konkreter Umsetzungen durch die Simulation Use Cases, welches dem Vorgehen im AP2.1 entspricht. Hierbei sollen die am „Proof of Concept“ beteiligten Simulation Use Cases speziell durch die durchgängige Dokumentation der Projektergebnisse unterstützt werden. Abbildung 91 zeigt die Top-Level-Sicht des Prozesses. Der Hauptprozess für den ersten Strang mit seinen In- und Outputs ist dabei oben dargestellt. Der zweite Strang befindet sich mit den zugehörigen In- und Outputs im unteren Teil der Abbildung. Beide sind durch die vorangestellte Entscheidung voneinander getrennt. Die Stränge können damit unabhängig voneinander durchlaufen werden, da keine direkten Abhängigkeiten bestehen.

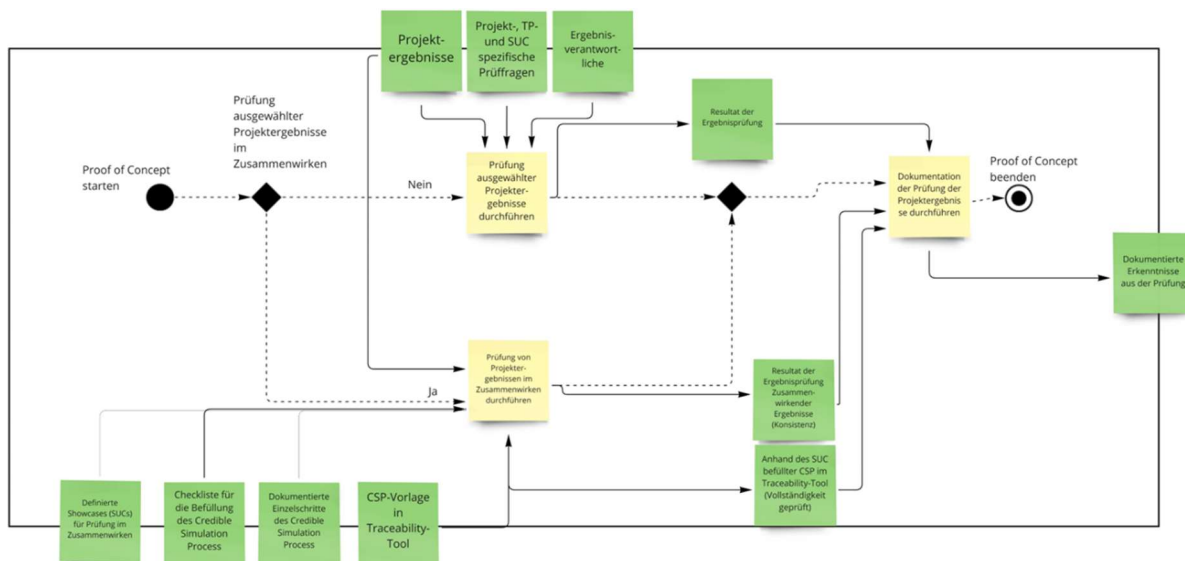


Abbildung 91: Top-Level-Sicht des Referenzprozesses zum „Proof of Concept“

Für die Durchführung des Prozesses zum Prüfen der Projektergebnisse werden als Inputs, die Projektergebnisse selbst deren Verantwortliche und die ergebnisspezifischen Prüffragen benötigt. Beim Durchlaufen des Prozesses wird entschieden, ob die Ergebnisse bereits reif genug für eine Prüfung sind. Zusätzlich wird entschieden ob gegebenenfalls noch weitere Prüffragen für das jeweilig zu prüfende Artefakt aufgenommen werden sollen. Die Entscheidung beziehungsweise Abstimmung darüber findet in Zusammenarbeit mit den Ergebnisverantwortlichen statt. Sie haben somit die Möglichkeit noch eigene Fragen zum Review beizusteuern. Mit Hilfe der daraus entstandenen Fragen wird der eigentliche Review durchgeführt. Im Anschluss werden die jeweiligen Ergebnisse des Reviews dokumentiert und damit der Prozess verlassen. Abbildung 92 zeigt die generische Darstellung eines Durchlaufs.

Für jeden Prozessschritt wurde zusätzlich eine exemplarische Beschreibung des Prozessschrittes mit dessen In- und Outputs erstellt. Dabei wurden für die In- und Outputs konkrete

Beispiele oder Erläuterungen für das Durchlaufen des Prozesses beschrieben. Abbildung 93 zeigt exemplarisch den Schritt „Durchführung der Prüfung auf Basis ausgewählter Fragen“ mit den Beispielen und Erläuterungen im unteren Teil.

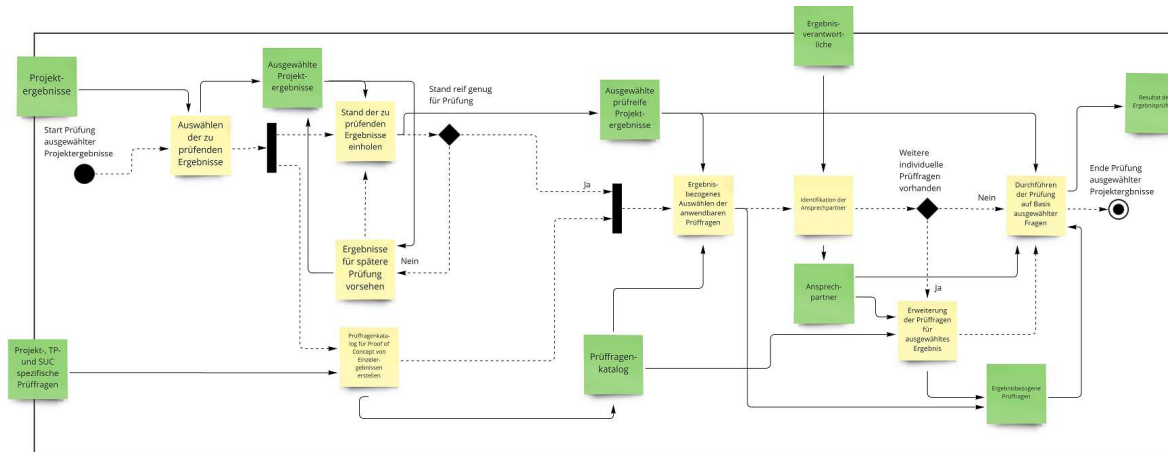
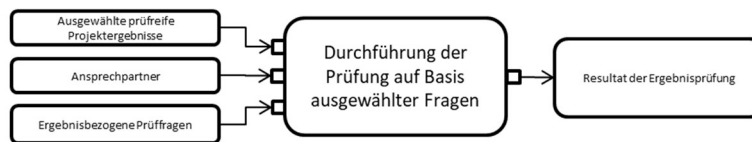


Abbildung 92: Prüfung einzelner Projektergebnisse



Input	Generisch	Spezifisch /Beispiel
	Ausgewählte prüf reife Projektergebnisse	Beispiel: Integrative Simulationsarchitektur
	Ansprechpartner	Verantwortliche für das jeweils zu prüfende Projektergebnis
	Ergebnisbezogene Prüfungen	Beispiel: „Haben die SUCs die integrative Simulationsarchitektur umgesetzt (um Anwendbarkeit zu zeigen)?“
Output	Generisch	Spezifisch /Beispiel
	Resultat der Ergebnisprüfung	https://gitlab.setlevel.de/pm/tp5/documents-tp5/-/tree/master/02_Durchfuehrungen/AP5.1/Reviewgegenstände
Detaillierte Aktivitäten (falls vorhanden)		

Abbildung 93: Exemplarische Beschreibung eines konkreten Prozessschrittes

Für die Durchführung des Prozesses zur Prüfung der „Projektergebnisse im Zusammenwirken“ werden als Inputs, die Simulation Use Cases, die Checkliste für den Credible Simulation Prozess, die dokumentierten Einzelschritte des Prozesses, sowie die Vorlage für diesen im Traceability Tool benötigt. Beim Durchlaufen des Prozesses werden die einzelnen Artefakte der SUCs anhand des Credible Simulation Prozess dokumentiert und fehlende Inhalte für die Durchgängigkeit identifiziert, was durch die Checkliste unterstützt wird. Zuletzt besteht die Auswahlmöglichkeit diese Artefakte und deren Zusammenhänge im Traceability Demonstrator toolgestützt abzubilden. Die einzelnen Schritte des Prozesses sind in Abbildung 94 dargestellt. Der Prozess stellt dabei das Vorgehen in der prozesssicheren Gesamtbeschreibung in AP2.1 dar. Die Anwendung wurde von AP2.1 gesteuert und in enger Zusammenarbeit mit den im Bild dargestellten Arbeitspaketen durchgeführt. Zur Vollständigkeit ist im Prozess auch die Entscheidungsmöglichkeit einer Weiterarbeit im Rahmen der industriellen Erprobung enthalten und der damit verbundene mögliche Übergang zum Einführungsprozess dargestellt.

Für diesen Prozess wurden ebenfalls wie zuvor beschrieben, für jeden Prozessschritt zusätzlich eine exemplarische Beschreibung des Prozessschrittes mit dessen In- und Outputs erstellt. Auch dabei wurden für die In- und Outputs konkrete Beispiele oder Erläuterungen gegeben.

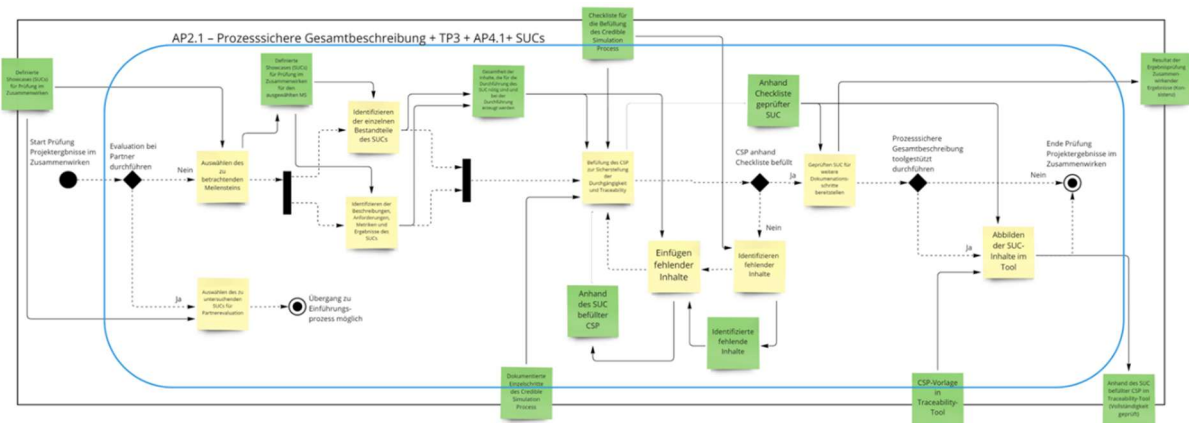


Abbildung 94: Prüfung von Projektergebnissen in deren Zusammenwirken

Für die Durchführung des „Proof of Concept“ wurden im SET Level Projekt TP- und Simulation Use Case spezifische Reviewfragen formuliert. Die Reviewfragen orientieren sich dabei an den jeweiligen projektspezifischen Ergebnissen aus den TPs beziehungsweise den Simulation Use Case sowie den im Projekt definierten Hauptergebnissen. Wie oben beschrieben wurden die unterschiedlichen Reviewkriterien zu einem Fragenkatalog zusammengeführt und für die Verwendung während des Reviews in einem Reviewtemplate für den jeweils angenommenen Reviewgegenstand zusammengefasst. Hierbei konnten auch ergebnisspezifische Fragen der Teilnehmer des Reviews noch mit aufgenommen werden. Abbildung 95 zeigt beispielhaft einen Auszug aus dem für die Reviews verwendeten Template. Die einzelnen Reviews wurden jeweils mit den Reviewverantwortlichen aus dem AP5.1 und den Ergebnisverantwortlichen basierend auf den erstellten Templates durchgeführt und die Ergebnisse in GitLab dokumentiert.

SET Level Review Form

1. **Gegenstand** <Bezeichnung, Beschreibung>
 1. **Datum/Version** des Gegenstandes
2. **Ansprechpartner** <Name(n) Ersteller/Verantwortliche >
3. **Quelle/Speicherort** <auch Dokumentation, wenn vorhanden>
4. **Review**
 1. Reviewer
 2. Datum des Reviews
5. **Detaillierte Beschreibung**
 1. **Darstellung des Zieles** <z. B. Funktionalitäten, Nutzbarkeit, ... >
 2. **Abweichungen vom Projektplan lt. VHB** <Änderungen in der Zielsetzung darlegen und kurz begründen >
 3. **Darstellung des Standes** < realisierter Umfang, Reifegrad, ggf. herausarbeiten, was noch fehlt>
 4. **Verfügbarkeit für Review** <Wie kann auf das Ergebnis zugegriffen werden; was wird für einen Review benötigt>
 5. **Materialien für Review** <zusätzliche Informationsquellen, z. B. Meilensteinbericht, Veröffentlichung>
6. **Planskizze zum Reviews** <etwa: Ausarbeiten der Fragen, Beantwortungsvorgehen, Beteiligte>
7. **Reviewfragen** <generisch>
 1. **Bewertung des Standes** in Bezug auf das Ziel
 2. **Mehrwert** gegenüber dem State of the Art und anderen Ansätzen
 3. **Einordnung in den Datenfluss** des Projektes <stimmig zu den anderen Ergebnissen (In-Out)>
 4. **Beitrag zu Projektzielen** <Richtung UAP 5.1.3>

Abbildung 95: Beispielhafte Darstellung des Reviewtemplates

PROSTEP führte dabei insbesondere anhand der erstellten Templates die Reviews zu den folgenden Artefakten durch:

- Simulation Use Case 1, 2 und 3
- Integrative Simulationsarchitektur
- Beispielhafte Werkzeugketten
- Traceability Demonstrator
- Credible Simulation Prozess

Basierend auf den Reviews sollen hier einige der wesentlichen Erkenntnisse exemplarisch erläutert werden. Die Durchführung der Simulation Use Cases ermöglichte eine erfolgreiche interne Erprobung verschiedenster Projektaktivitäten. Dazu zählen unter anderem die aktive Nutzung und Integration verschiedener im Projekt entwickelter Modelle, sowie die Anwendung des Credible Simulation Process. Zusätzlich konnte für die Simulation Use Cases 2 und 3 die Übertragbarkeit der Ansätze auf unterschiedliche Werkzeugketten gezeigt werden. Dies wurde auch im Review einer exemplarischen (vendorspezifischen) Werkzeugkette bestätigt. Zusätzlich wurden bei der Umsetzung der Simulation Use Cases verschiedenste Standards verwendet. Hier sind beispielweise OSI, FMI und die Open Standards, OpenDRIVE und OpenSzenario zu erwähnen. Damit konnte auch die Einsetzbarkeit dieser Standards untersucht und sichergestellt werden. Insbesondere wurde eine OSI-Kommunikation in allen Simulation Use Cases als Schnittstelle verwendet.

Mit der integrativen Simulationsarchitektur konnte eine durchgängige Methodik zur Herleitung der Architekturen von konkreten Simulationsläufen aus z.B. der Architektur des realen hochautomatisierten Fahrzeugs erstellt und eine Anknüpfung an den Credible Simulation Process geschaffen werden. Sie unterstützt die Fehlerminimierung und Automatisierung z.B.: durch die Ausleitung von SSP-Dateien für Simulationssoftware. Durch die Verknüpfung der einzelnen Simulationsläufe mit der Architektur des realen Fahrzeugs gemäß SAFAD-Paper [13] ist die Übertragbarkeit von Simulationsaussagen auf die Realität sichergestellt. Der industrielle Mehrwert der Architektur wird zudem durch den ihr zu Grunde liegenden MBSE-Gedanken unterstützt. Der dabei umgesetzte RFLP-Ansatz bietet eine gute Basis für die Diskussion über die einzelnen Architekturelemente, da die einzelnen Ebenen eindeutig voneinander abgegrenzt werden können. Speziell die Einbeziehung des gesamten Systems inklusive seiner Umgebung trägt dabei dem Trend vom klassischen Ansatz hin zum software-defined Vehicle Rechnung, da damit die deutlich zunehmenden Interaktions- und Kommunikationspunkte mit der Umgebung erfasst werden können.

Im Rahmen einer beispielhaften Werkzeugkette konnte gezeigt werden, dass sich die OSI-Kommunikation für die Zusammenschaltung unterschiedlicher im Projekt entwickelter Modelle anwenden lässt. OSI definiert sowohl semantische und technische Ebene. Dadurch, dass die Modelle über eine OSI-Kommunikation angebunden werden können, wird dabei die Anforderung für die Komposition auf semantischer Ebene und technischer Ebene erfüllt. Hinsichtlich der Anwendung der Architektur ergibt sich eine Einschränkung. Die im Projekt entwickelte Architektur findet in den Simulation Use Cases Anwendung und die Vendorswerkzeugketten sind darin eingebunden. Allerdings wird eine einheitliche Architektur über beziehungsweise innerhalb verschiedene Vendors tools als nicht sinnvoll angesehen.

Der Mehrwert in der Anwendung des Credible Simulation Process liegt in der Schaffung eines geleiteten Ansatzes, um das aktuell häufig dokumentenzentrierte Vorgehen durchgängig und prüfbar in einem konsistenten Prozess abzubilden. Dies konnte auch durch seine Anwendung in den Simulation Use Cases gezeigt werden. Zusätzlich konnte ein industrieller Mehrwert durch das Mapping zwischen den Phasen des Credible Simulation Process und der Simulationsarchitektur identifiziert werden. Insbesondere die Möglichkeit der Ausleitung als SSP wird dabei als ein großer Mehrwert angesehen, da durch die Erzeugung von Metamodellen die Wiederverwendbarkeit signifikant erhöht wird und der Wiederverwendungsgedanke nicht nur abstrakt, sondern zentral verankert ist. Zusätzlich kann die Anwendung des Credible Simulation Process durch den im Projekt entwickelten Traceability Demonstrator toolgestützt unterstützt werden. Diese Unterstützung wird durch ein im Tool hinterlegtes Template vereinfacht. Ein

weiterer Mehrwert, auch im Zusammenspiel mit dem Credible Simulation Process, entsteht durch die Reduzierung der Komplexität durch die Schaffung eines Informationsnetzwerks über bestehende Systeme und Datenbestände. Dazu schafft das Tool eine Verlinkung von Artefakten und persistiert Abhängigkeiten. Die Anwendbarkeit konnte im Rahmen verschiedener Dokumentationen der Simulation Use Cases anhand des Credible Simulation Process gezeigt werden.

Nach dem formalen Ende der Arbeiten zur „Prozesssicheren Gesamtbeschreibung“ im AP2.1 wurde dieses Thema noch einmal im AP5.1 aufgegriffen. Da die Simulation Use Cases für den „Proof of Concept“ verwendet wurden, liefert auch deren durchgängige Beschreibung anhand des in SET Level entwickelten Prozesses und die darauf aufbauende Abbildung im Traceability Tool einen Beitrag zur internen Erprobung und kritischen Reflexion der Ergebnisse. Im Rahmen der erneuten Aufnahme der Arbeiten konnten damit auch die aktuellsten Stände der Dokumentation der Simulation Use Cases erfasst werden. Zusätzlich war die Wiederaufnahme der Arbeiten dadurch bedingt, dass im Traceability Tool ein neues Datenmodell zur Verfügung stand. Es erfolgte daher eine Überarbeitung der in GitLab zur Verfügung gestellten Dokumentation der Simulation Use Cases im Traceability Tool. Die im Tool neu hinzugekommenen Fähigkeit des Ziehens von konkreten Baselines konnte dabei für verschiedene Meilensteine angewendet werden. Die im Traceability Tool durchgeführte Dokumentation wurde in XML und als zip ausgeleitet und auf GitLab für die jeweiligen Simulation Use Cases abgelegt. Dies ermöglicht den Projektpartnern auch nach Projektende, den Zugriff auf die Dokumentation in einem neutralen Austauschformat. Abbildung 96 zeigt exemplarisch einen Subschritt des Credible Simulation Process nach dem die Dokumentation für den Simulation Use Case 1 im Tool mit dem neuen Datenmodell durchgeführt wurde.

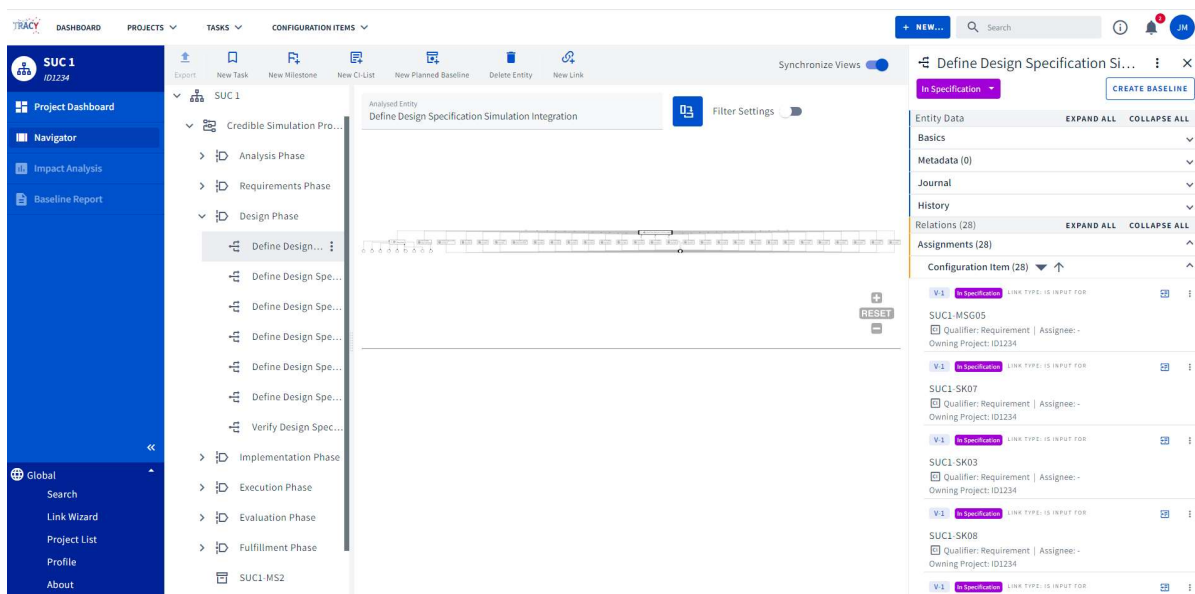


Abbildung 96: Dokumentation des Simulation Use Case 1 im Traceability Demonstrator anhand des Credible Simulation Process (ausgewählter Subschritt)

1.1.5.2 AP 5.2: Informationsgewinn aus den Erkenntnissen der Industriepartner aus der firmenspezifischen Erprobung

Generischer Einführungsprozess

Mit Hilfe des generischen Einführungsprozesses für SET Level-Artefakte, wird es allen am SET Level-Projekt beteiligten Unternehmen ermöglicht, die in den Simulations-Use-Case (SUCs) erstellen Artefakte (Modelle, Parametersätze, Prozesse, Methoden etc.), im Firmenumfeld einzuführen und zu erproben. Des Weiteren bietet der generische Einführungsprozess den beteiligten Unternehmen die Möglichkeit, die Erkenntnisse, die sie bei der firmeninternen Erprobung gewonnen haben, dem Projekt zurückzugeben und so den Entwicklungs- und Verbesserungszyklus der Artefakte zu schließen.

Der generische Einführungsprozess für SET Level-Artefakte im firmenspezifischen Kontext ist in die nachfolgenden sieben Haupt-Prozessschritte unterteilt:

- Auswahl und Mapping der für den Simulations-Use-Case notwendigen Artefakte auf vorhandene Artefakte im Unternehmen (Modelle, Werkzeuge, Werkzeugketten, Schnittstellen, Parameter)
- Identifikation der für den SET-Level Simulations-Use-Case fehlenden firmeninterne Artefakte (Modelle, Werkzeuge, Werkzeugketten, Schnittstellen)
- Aufbau der Simulationsumgebung für den Simulations-Use-Case
- Durchführung / Ausführung des Simulations-Use-Cases
- Aufbereitung und Dokumentation der Ergebnisse
- Aufgetretene Fehler dokumentieren
- Durchführung der Nachverfolgbarkeit (Traceability) der Entscheidungen, der eingesetzten Modelle, Werkzeuge, Werkzeugketten, Schnittstellen etc. für Simulations-Use-Case

Jedem Prozessschritt sind spezifische Ein- und Ausgabe-Artefakte zugeordnet. Diese werden von dem Prozessschritt als Input benötigt, bzw. als Output generiert. Der generische Einführungsprozess kann für alle im Projekt definierten SUCs und den darin benötigten Artefakten gleichermaßen eingesetzt werden. Der generische Einführungsprozess muss dafür nicht angepasst werden. Prinzipiell ist der Einführungsprozess derart generisch gehalten, dass auch andere Simulationen damit eingeführt werden können.

Die Abbildung 97 zeigt den generischen Einführungsprozess, bestehend aus den obigen Prozessschritten, inklusive aller Ein- und Ausgabe-Artefakte und Entscheidungen, auf der obersten Ebene. Die Darstellung des Prozesses erfolgt als UML/SysML-Aktivitätsdiagramm.

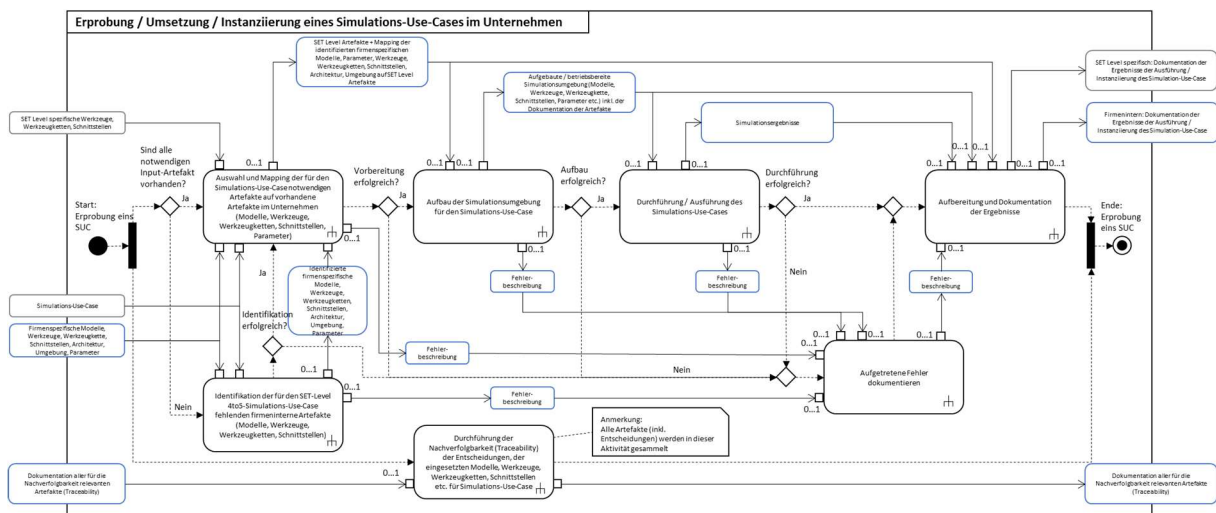


Abbildung 97: Der generische Einführungsprozess für SET Level-Artefakte

Für alle Prozessschritte sind die Ein- und Ausgaben (Artefakte), unterteilt in SET Level-spezifische (blauer Rahmen) und firmenspezifische Artefakte (grauer Rahmen), explizit modelliert und mit einer Kardinalität versehen. Die Beschreibung der Ein- und Ausgabe-Artefakte ist zum einen direkt am Ein- bzw. Ausgang („Port“) des Prozessschritts angegeben und zum anderen in einer Tabelle beschrieben. Dabei geht die tabellarische Beschreibung über die Beschreibung der Artefakte im Prozess hinaus, z.B. um die für einen spezifischen SUC benötigten Eigenschaften von Artefakten oder die Verantwortlichkeiten für ein spezifisches Artefakt.

Zur Vereinfachung der firmenspezifischen Anwendbarkeit wurde im Rahmen des Projekts entschieden, den allgemeinen generischen Einführungsprozess, ohne den Prozessschritt für die Fehlerdokumentation, zu verwenden. Des Weiteren wurden einige Entscheidungspunkte entfernt. Die Abbildung 98 zeigt den vereinfachten generischen Einführungsprozess für SET

Level-Artefakte aus der Abbildung 97, ohne den Prozessschritt der Dokumentation und den zusätzlichen Entscheidungen (vgl. X1 - X5 und X7).

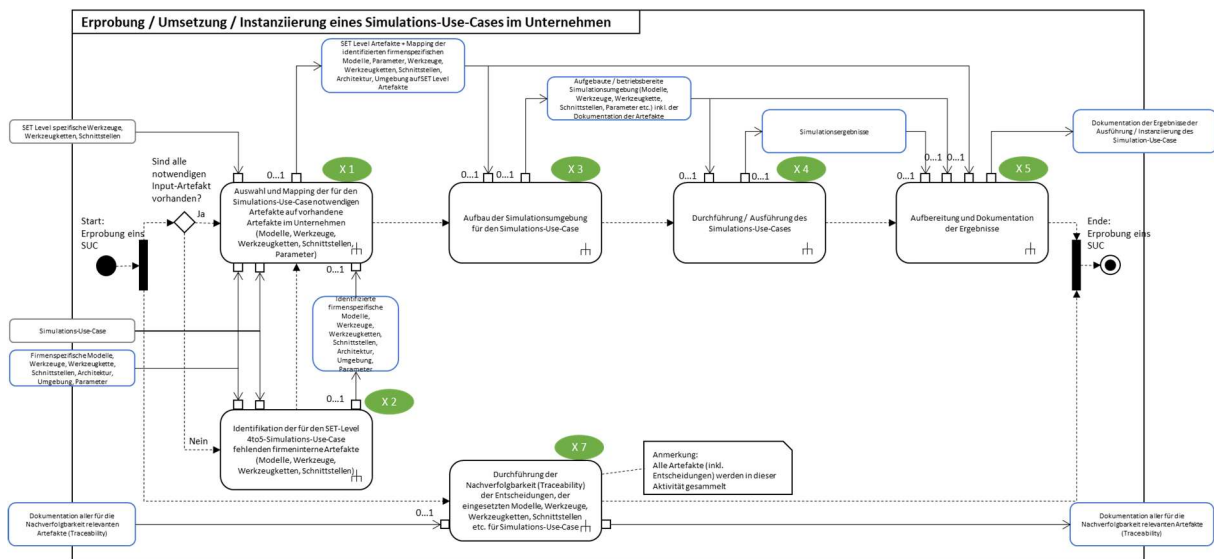


Abbildung 98: Vereinfachte Darstellung des generischen Einführungsprozesses

Anforderungserhebung

Abbildung 99 zeigt schematisch das gesamte Vorgehen von Erhebung bis Priorisierung der Anforderungen die aus Sicht der in SET Level beteiligten Industriepartner eine hohe Priorität im Hinblick auf automatisiertes Fahren haben.

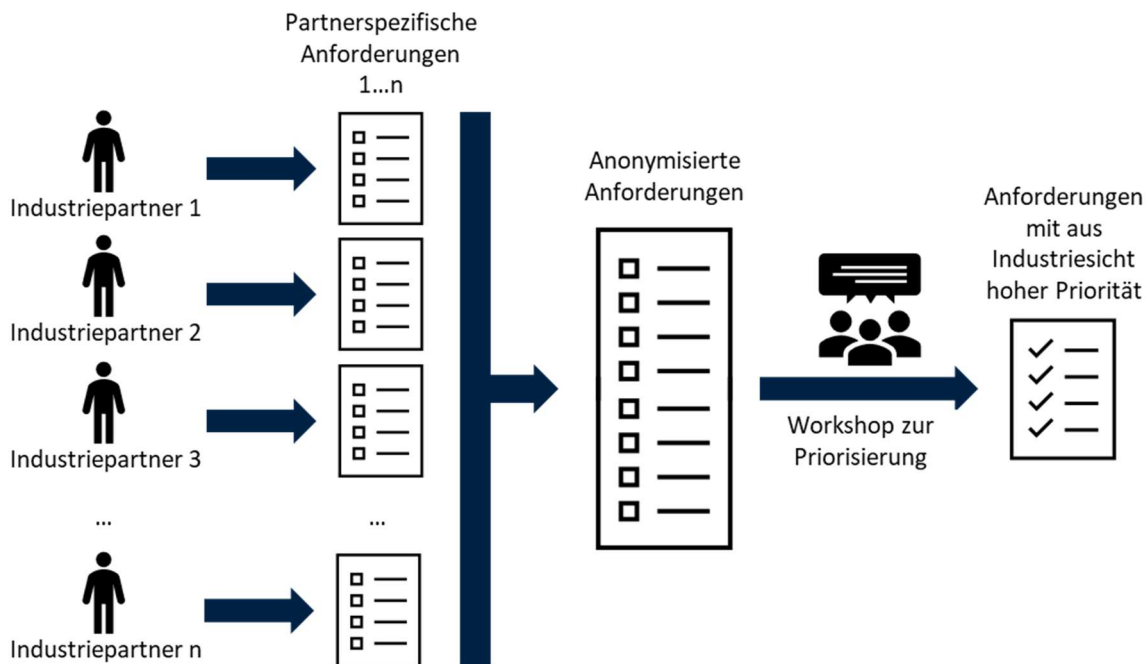


Abbildung 99: Vorgehen zur Ermittlung von Anforderungen die aus Industriesicht eine hohe Priorität haben

Für die Erhebung der Anforderungen aus der Industrie wurden im Projekt SET Level von PROSTEP separate Workshops einzeln mit den jeweiligen Industrievertretern in den Häusern durchgeführt. Dabei fand eine Erhebung des aktuellen IST-Zustandes sowie des angestrebten SOLL-Zustandes in Bezug auf SET Level relevante Themengebiete statt.

Dieses Vorgehens verfolgte dabei mehrere Ziele. Zum einen konnte dadurch der firmeninterne Austausch innerhalb der beteiligten Fachabteilungen gefördert und ein Bewusstsein für die jeweiligen aktuellen Stände, eingesetzten Tools, verwendeten Methoden und Prozesse der

Abteilungen geschaffen werden. Zum anderen konnte ein gemeinsames Bild, der teils innerhalb der Abteilungen unterschiedlichen die Erwartungen und Ziele erarbeitet werden. Diese gemeinsame Erarbeitung des SOLL-Zustandes förderte auch das Problembewusstsein und -verständnis unter den Beteiligten über Abteilungsgrenzen hinweg. Darüber bildeten die Inhalte des SOLL-Zustandes die Grundlage für die nachfolgende partnerübergreifende Anforderungserhebung. Abbildung 100 zeigt exemplarisch einen Teil der aufgenommenen Inhalte eines Workshops. Die konkreten Inhalte sind aus Vertraulichkeitsgründen unkenntlich gemacht.



Abbildung 100: Exemplarische Ergebnisse eines Workshops (aus Vertraulichkeitsgründen unkenntlich gemacht)

Die auf diesem Wege erarbeiteten Inhalte wurden aufbereitet und in ein für diesen Zweck definiertes Datenmodell in Cameo Systems Modeller überführt. Hierbei sind die Inhalte der einzelnen Partner für die spätere Ausleitung wieder getrennt im Modell abgebildet. Die innerhalb des Modells dokumentierten Artefakte wurden jeweils partnerspezifisch neutralisiert, zum Beispiel durch Entfernung von Produkt- oder Projektnamen aus den Inhalten, die einen direkten Rückschluss auf den jeweils beteiligten Partner zulassen.

Daran anschließend fand eine für jeden Partner separat durchgeführte Freigabe der neutralisierten Ergebnisse statt um sicherzustellen, dass keine Inhalte in das Projekt zurückgespielt werden, die vom jeweiligen Partner als intern oder vertraulich eingestuft werden. Abbildung 101 zeigt die Grobstruktur des im Cameo Systems Modeller definierten Datenmodells.

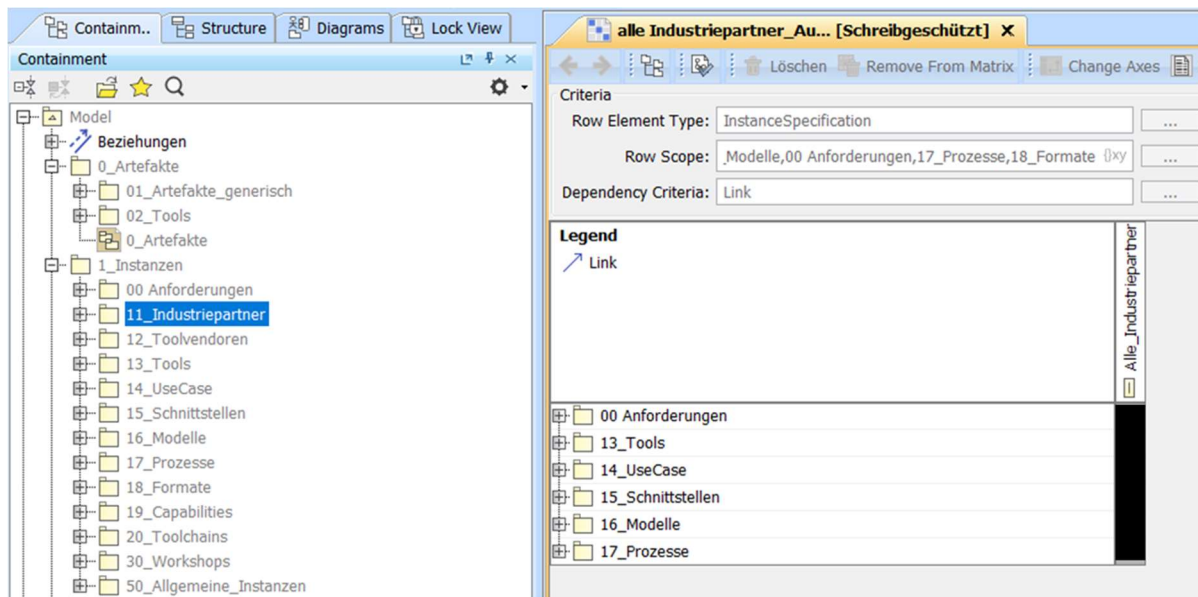


Abbildung 101: Grobe Struktur Cameo

Im Anschluss an die Verifikation der Ausleitungen der neutralisierten Auswertungen der Workshops mit den Industriepartnern wurden diese für eine partnerübergreifende Priorisierung aufbereitet. Ziel ist auch bei einer gemeinsamen Betrachtung der erarbeiteten Inhalte keine direkten Rückschlüsse auf Aussagen, Probleme und Erwartungen der jeweiligen Industriepartner zu ermöglichen. Insgesamt wurden bei diesem Vorgehen mehr als 400 Inhalte gemeinsam mit allen beteiligten Industriepartnern erarbeitet.

Die Gesamtheit der Inhalte wurde themenspezifisch kategorisiert und nach IST-Zustand und SOLL-Zustand unterschieden, für die partnerübergreifende Priorisierung ausgeleitet. Abbildung 102 zeigt einen Ausschnitt aus dieser Ausleitung. Die Inhalte sind aus Vertraulichkeitsgründen unkenntlich gemacht.



Abbildung 102: Ausleitung (aus Vertraulichkeitsgründen unkenntlich gemacht)

Basierend auf der neutralisierten Ausleitung wurde die Priorisierung der Anforderungen in gemeinsamen, übergreifenden Workshops zusammen mit den im Projekt involvierten Firmenvertretern der jeweiligen Industriepartner durchgeführt. Generell lassen sich aus der Anforderungserhebung und der durchgeführten Priorisierung, die Anforderungen und Erwartungen der Industriepartner an das Projekt wie folgt formulieren:

Die Erwartung aus der Industrie an das Projekt ist, anwendbare Werkzeuge und Methoden für Szenarien und Testcases bereitstellen sowie eine breite Unterstützung von Standards in Simulationstools zu gewährleisten.

Die Ergebnisse der Priorisierung wurden auf GitLab abgelegt und damit dem Projekt für alle Partner zugänglich zur Verfügung gestellt. Aufbauend auf den in den Industrieworkshops erhobenen Anforderungen und der anschließenden Priorisierung, im Hinblick auf automatisiertes Fahren, fand eine Generalisierung der Anforderungen sowie eine Einteilung nach Themenclustern statt. Die durchgeführte Generalisierung und Zuordnung wurde mit den im Projekt involvierten Industriepartnern abgestimmt. Die Ergebnisse der Arbeiten sind in Abbildung 103, nach Themenclustern sortiert, abgebildet.

Im Rahmen der Generalisierung, wurden dabei auch unten Einbeziehung der Ergebnisse des IST-Zustandes die Anforderungen an das Projekt bezugnehmende auf die Themencluster konkretisiert. Darüber hinaus erfolgte ein initiales Mapping der generalisierten und konkretisierten Anforderungen auf die einzelnen Arbeitspakete und Arbeitsgruppen des SET Level Projektes, wobei keine Widersprüche zu den Projektzielen identifiziert, werden konnten, da in den Anforderungen die aus Industriesicht hohe Priorität für automatisiertes Fahren haben, auch Anforderungen enthalten sind die klar nicht im Scope des SET Level Projektes liegen. Die Ergebnisse wurden im Rahmen eines Quartalsworkshop dem Projekt übergeben und für alle Partner zugänglich auf GitLab abgelegt. Die daraus resultierenden Handlungsschwerpunkte wurden im Rahmen des Workshops kommuniziert und an das Projekt übergeben.

SET Level Industriepartner

Themencluster	Szenarien	Methoden	Modelle	Validierung	Standardisierung	Testen
Anforderungen	Identifikation kritischer Szenarien	Sicherstellung Zulassungsfähigkeit	Validierung von Modellen und Validierungsmetrik für Modelle		KPI	Testen der AD-Funktion
	Kreuzungsszenario mit Spurwechsel	Toolqualifizierung für virt Simulation	Toolübergreifende Modelle	Validierte Verkehrsteilnehmer	Standardisierte Testdefinition	
	Szenariobeschreibungssprache	Modellentwicklung und Simulation		Validierte Schnittstellen	Schnittstellen (HiL, SiL und Sensormodelle)	Testen von Sensoren und Komponenten
	Szenariobeschreibung und Handling		Validierung von (Sensor-)modellen		FMI, OSI, OSC, ODR, OpenSCENARIO	Definierter Testprozess
			Freigabe und Spezifikation (für Sensormodelle)		Standardisierte modulare Simulationsarchitektur	
			Rohdaten für Sensormodelle			

Abbildung 103: Aus Industriesicht priorisierte Anforderungen an automatisiertes Fahren, geordnet nach Themenclustern

Diskussionsplattform für die Industrie

Zur Unterstützung der industriellen Anwendbarkeit der Projektergebnisse wurde für ein Prozess für den kontinuierlichen Austausch und Review der Projektergebnisse erarbeitet und mit den TP-Leitern abgestimmt. Die daraus resultierende Diskussionsplattform mit den Industriepartnern ermöglicht es, den Projektpartnern konkrete Projekt- oder Zwischenergebnisse zur Diskussion zu stellen, und mit den Industrievertretern abzustimmen. Dabei existieren prinzipiell zwei Pfade. Einerseits kann Feedback direkt innerhalb dieser Runden gegeben werden. Andererseits besteht die Möglichkeit, dass Feedback aus Vertraulichkeitsgründen an das TP5 übergeben wird und dann in neutralisierter Form durch das TP5 eingebracht wird.

Im Rahmen der Diskussionsplattform wurden die Themengebiete Anforderungen, Architektur, Schnittstellen, sowie SRMD/Metadaten für Modelle behandelt. Darüber hinaus wurden die initialen Abstimmungsrunden für die anschließende industrielle Erprobung der SET Level Kooperationsprozesse mit dem SmartSE Referenzprozess [1] zum Modellaustausch zusammen mit dem TP3 durchgeführt.

Industrielle Erprobung

Ziel der industriellen Erprobung von SET Level Ergebnissen war es, die industrielle Anwendbarkeit der Ergebnisse zu unterstützen. Dazu wurden verschiedene Projektergebnisse einem Review durch die SET Level Industrievertreter unterzogen. Trotz des teilweisen unterschiedlichen Vorgehens im Rahmen der Reviews bzw. der Erprobung wurde grundsätzlich darauf geachtet, dass partnerspezifische Inhalte, soweit dies gefordert war, nur neutralisiert und nach Freigabe der Partner in das Projekt oder weitere Diskussionen eingebracht wurden. Damit sollte sichergestellt werden, dass keine Partnerinterna in das Projekt fließen.

Kooperationsprozess

Basierend auf den initialen Abstimmungsrunden im Rahmen der Industrieplattform, wurde in TP5 zusammen mit dem TP3 ein Konzept entwickelt, um einen Review des Zusammenspiels der Kooperationsprozesse zusammen mit den SET Level Industriepartnern durchzuführen. Hierzu wurde zusätzlich zu den in TP3 erarbeiteten Prozessen, Credible Simulation Process und Credible Modelling Process der in der prostep ivip Arbeitsgruppe SmartSE entwickelte Referenzprozess [1] mit eingebracht.

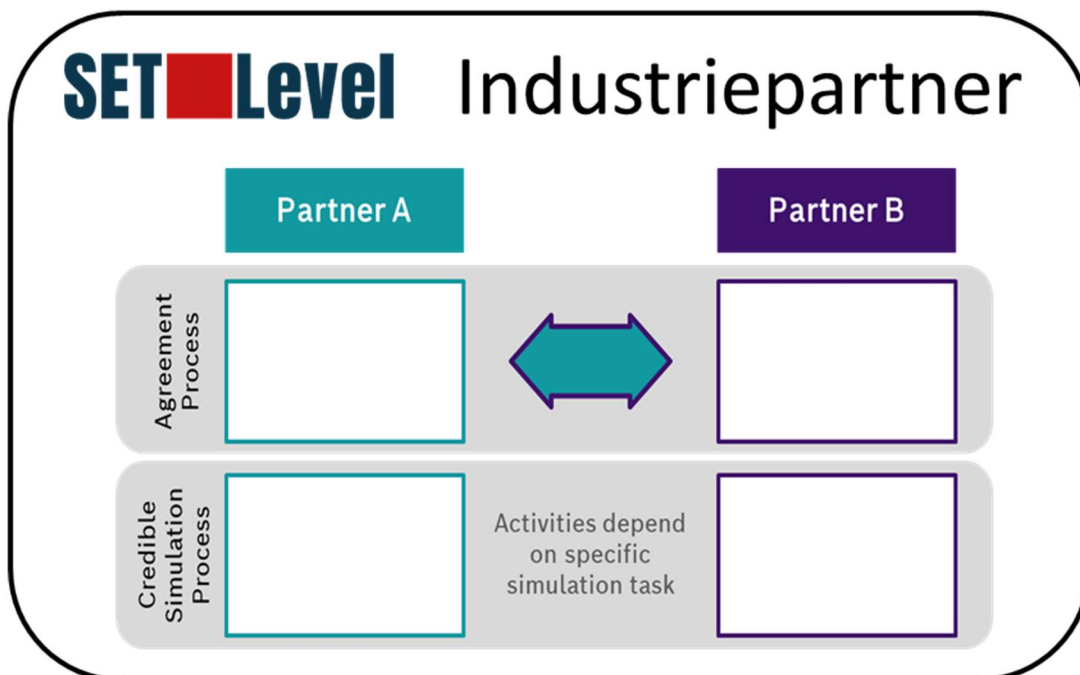


Abbildung 104: Abbildung der Hauptphasen des Kooperationsprozesses in Anlehnung an [1], beteiligt Partner aus Vertraulichkeitsgründen nicht einzeln aufgelistet

Hintergrund der Definition des Kooperationsprozesses in TP3 und der industriellen Erprobung der Anwendbarkeit des Prozesses in TP5, ist der wachsende industrielle Trend hin zu gemeinsamen agilen Entwicklungen, was insbesondere im Bereich des automatisierten Fahrens gilt. Beispielhaft kann hier die gemeinsame Entwicklung einer automatisierten Fahrfunktion, die ein Kameramodell, als Wahrnehmungsanteil der Gesamtfunktion beinhaltet angeführt werden. Dieser Use Case zeigt die Notwendigkeit einer engen Zusammenarbeit und Koordination zwischen den beteiligten Partnern, im Gegensatz zu früheren Ansätzen, bei denen das Kameramodell nach vorgegebener Spezifikation von einem Partner alleine entwickelt wurde.

Ein Kooperationsprozess besteht in der Regel aus einem Agreement-Prozess und einem Entwicklungsprozess. Er beginnt mit dem Agreement-Prozess, der zur Klärung und Festlegung der technischen und vertraglichen Rahmenbedingungen dient. Nachdem der Agreement-

Prozess abgeschlossen ist, beginnt der eigentliche Entwicklungsprozess, der die gemeinsame Spezifikation, das Design und die Implementierung der Simulations- und Modellierungsaufgaben umfasst. In dem für die Erprobung in SET Level definierten Falle wird der Entwicklungsprozess durch Credible Simulation Process und Credible Modelling Process repräsentiert. Abbildung 104 zeigt die beiden Hauptphasen des Kooperationsprozesses.

Um die industrielle Anwendbarkeit des Prozesses zu gewährleisten und sicherzustellen, dass der Prozess die Zusammenarbeit zwischen Industrieunternehmen unterstützt, wurden mehrere Workshops mit den Industrievertretern (siehe Abbildung 104) des SET Level Projektes durchgeführt. Ziel dieses Vorgehens, war es einerseits den Prozess im Allgemeinen zu diskutieren und ein gemeinsames Verständnis zu schaffen. Andererseits sollte der Prozess durch die beteiligten Industriepartner in einer möglichst realitätsnahen Konstellation durchgespielt werden. Dazu fanden die Workshops jeweils separat mit Paaren aus OEM und Zulieferer statt. Die Zusammenarbeit und Dokumentation fanden online über MS-Teams und das Tool Miro statt.

Die Ergebnisse der Einzelworkshops wurden jeweils für die beiden Partner separat erfasst und aufbereitet. Basierend auf der aufbereiteten Rückmeldung an die Einzelgruppen fand durch diese, eine Freigabe statt um die gewonnenen Erkenntnisse in der gesamten Gruppe zu teilen und weiter diskutieren zu können. Insgesamt wurden dabei zusammen mit allen Partnern über 300 individuelle Rückmeldungen eingeholt. Für die weiterführende Bearbeitung fand eine Überführung und Konsolidierung in ein gemeinsames Miro-Board statt um eine weitere partnerübergreifende Zusammenarbeit zu ermöglichen. Ein Ausschnitt des Ergebnisses ist in Abbildung 105 dargestellt. Die genauen Inhalte sind aus Vertraulichkeitsgründen weichgezeichnet.

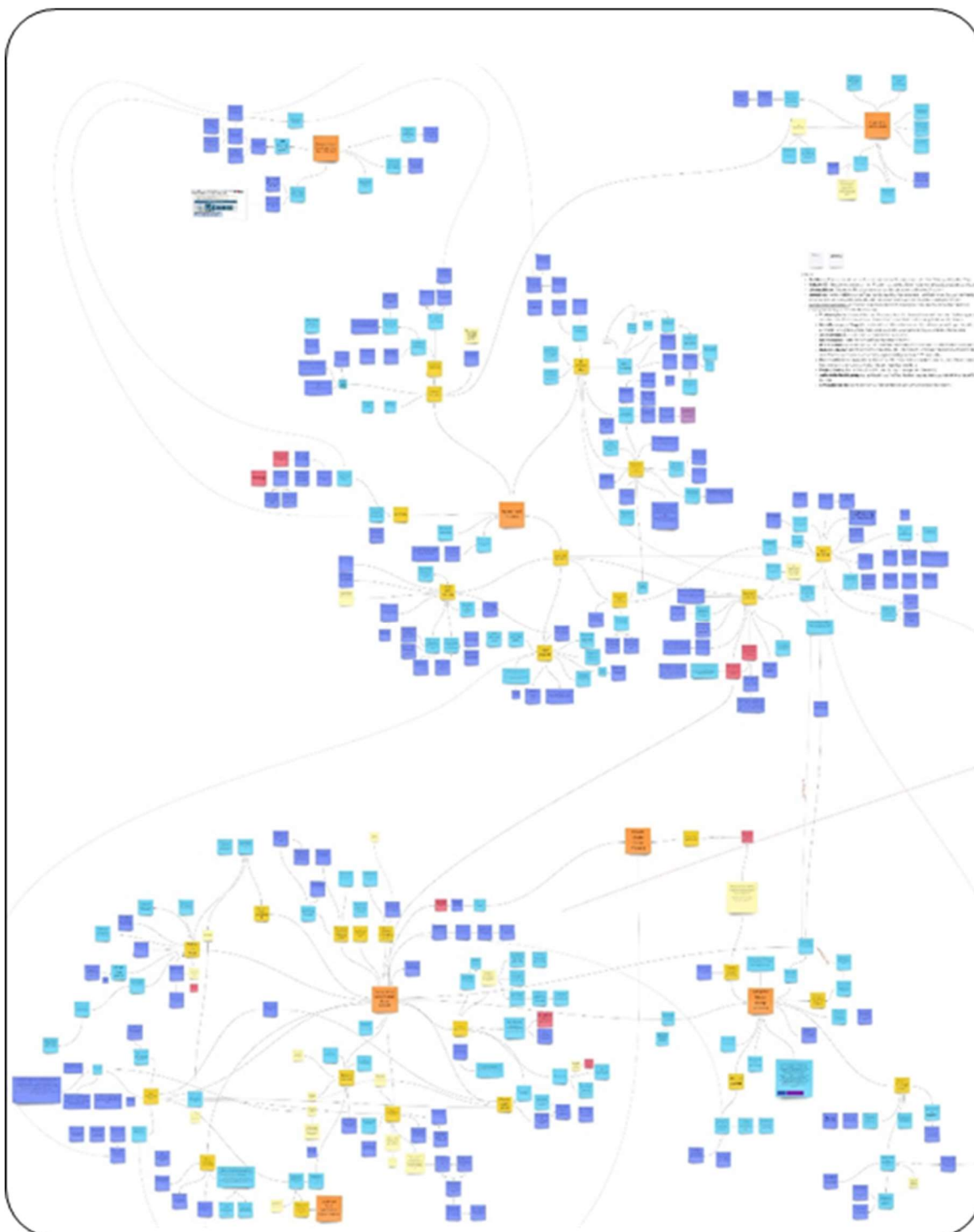


Abbildung 105: Auszug aus den konsolidierten Ergebnissen der Workshops (aus Vertraulichkeitsgründen weichgezeichnet)

Basierend auf den konsolidierten Ergebnissen konnte mit den SET Level Partnern eine Priorisierung der Erkenntnisse, in Bezug auf Zusammenarbeit und den Austausch zwischen Projektpartnern im realen Projektumfeld durchgeführt werden. Im Rahmen der Workshops und der anschließenden Priorisierung konnten jedoch nicht nur Herausforderungen, die sich durch die partnerübergreifende Zusammenarbeit bei der Entwicklung hochautomatisierter Systeme ergeben identifiziert werden, sondern es wurden auch darauf aufbauend Best Practices und Recommendations abgeleitet.

Nach Beendigung der Workshops und der gemeinsamen Priorisierung wurden die Ergebnisse an die Verantwortlichen im TP3 übergeben und diese bei der Erstellung eines Dokuments, das

speziell die Herausforderungen und Best Practices beschreibt, unterstützt. Darüber hinaus wurde über das Vorgehen und die Ergebnisse im Rahmen eines Vortrags auf dem SET Level Abschlussevents berichtet und zusammen mit dem TP3 und Industrievertretern die Möglichkeit in Form eines Marktstandes geschaffen, die Diskussion in die Breite zu tragen.

Simulation/ Modelling Request

Eine wichtige generelle Erkenntnis, die aus der industriellen Erprobung des SET Level Kooperationsprozesses gewonnen wurde, war das für den Austausch von Simulationsmodellen und der partnerübergreifenden Zusammenarbeit im Rahmen von Simulationen und deren Aufbau, ein einheitlicher Satz an Metadaten für die Übergabeartefakte definiert sein sollte. Dies unterstützt einerseits ein standardisiertes Vorgehen bei der Beauftragung von Simulationsaufgaben, sowohl innerhalb des Unternehmens als bei der partnerübergreifenden Zusammenarbeit. Der im TP3 entwickelte Simulation Request und Modeling Request definiert dazu ein Framework, in dem auch ein standardisiertes Datenmodell für den Austausch definiert wurde. Der Simulation Request bzw. Modelling Request ist dabei der Ausgangspunkt für eine Absprache zwischen zwei Partnern, um eine Simulation zu beauftragen. Die im Simulation Request bzw. Modelling Request enthaltenen Informationen dienen als Input des CSP (siehe Abbildung 106). Das zugehörige neutrale Austauschformat leitet sich aus dem bestehenden STMD (Simulation Task Meta Data) ab.

Darauf basierend wurde in einer ähnlichen Konstellation wie für die Erprobung der Kooperationsprozesse ein industrieller Review der Requests durchgeführt. Dazu wurden zuerst partnerübergreifende Workshops mit den Industrievertretern und den Verantwortlichen und Partnern aus TP3 durchgeführt. Daran anschließend wurden die in den gemeinsamen Workshops gewonnenen Erkenntnisse in separaten Workshops mit jeweils einem OEM und Tier detaillierter in Bezug auf den jeweiligen Anwendungsfall diskutiert, um den in der Realität auftretenden Zusammenarbeitsmodus nachzubilden und die Anforderungen an eine reale Umsetzung des Requests im Zusammenspiel eines Auftraggebers und Auftragnehmers durchzuspielen.

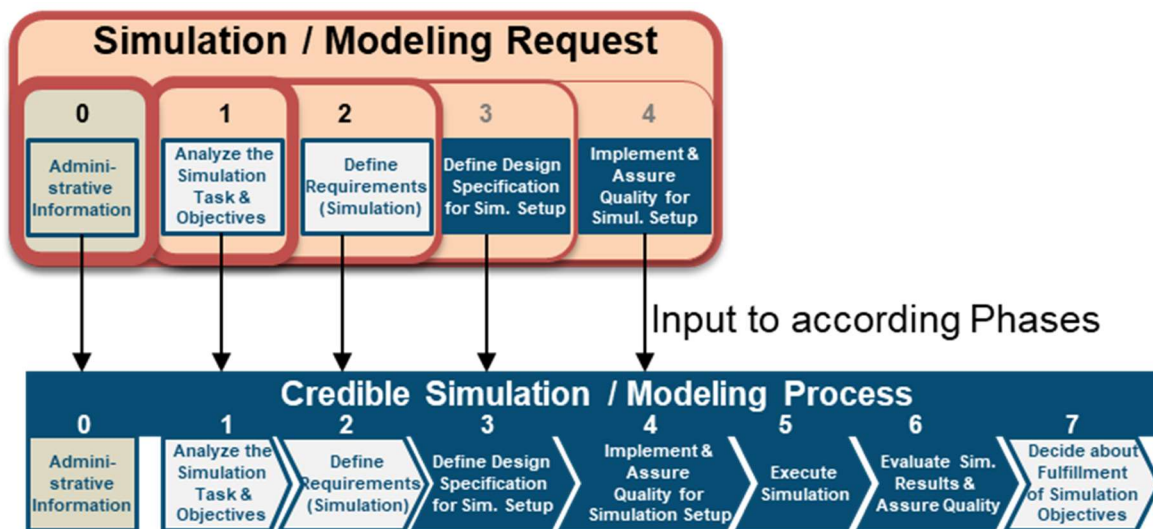


Abbildung 106: Zusammenhang zwischen Simulation / Modelling Request und Credible Simulation / Modelling Process

Bei den Reviews lag der Schwerpunkt hierbei auf den ersten drei Teilen des Simulation Request bzw. Modelling Request, die wie in Abbildung 106 dargestellt, den ersten Phasen des Credible Simulation und Credible Modelling Process zugeordnet werden können.

Einige der Haupterkenntnisse des Reviews und aus der Diskussion der Requests sind, die Identifikation von zwingend notwendigen Inhalten, die im Datenschema der Requests für den Austausch enthalten sein müssen, um ein vereinheitlichtes minimales Set an Metadaten zur Unterstützung des Kooperationsprozesses mit optionalen Bausteinen zu definieren. Darüber hinaus existiert Konsens darüber, dass bestimmte Elemente der Request automatisch zu

befüllen sein müssen und dies auch für den Nutzer explizit hervorgehoben werden sollte, um eine größtmögliche Vereinheitlichung sicherzustellen. Für die sinnvolle einheitliche Nutzung der Requests als standardisiertes Dokumentationsschema zur Unterstützung der Traceability im STMD-Format ist es darüber hinaus zwingend notwendig, die Inhalte bereits frühzeitig in der Agreementphase abzustimmen, klar zu definieren und die Ergebnisse auch in einem in der Agreementphase erstellen Glossar festzuhalten und allen Beteiligten zur Verfügung zu stellen.

Zusätzlich konnten innerhalb der Partnerworkshops jedoch auch einige Einschränkungen identifiziert werden. Aktuell ist nicht davon auszugehen, dass ein OEM eine gesamte Simulation beauftragt, sondern nur eine Modellerstellung. Es ist aber nicht ausgeschlossen, dass sich dies in Zukunft ändern kann. Damit existiert an der Schnittstelle zwischen OEM und Lieferant primär nur das Modelling Request. Wobei das Modelling Request jedoch prinzipiell in beide Richtungen gestellt werden (z.B.: wenn ein Lieferant eine Modellbeisteuerung vom OEM benötigt). Das Simulation Request kann jedoch firmenintern z.B. beim OEM als Leitfaden und Dokumentationsschema, welche Informationen der Auftragnehmer vom Auftraggeber benötigt, um eine Simulation durchführen zu können, verwendet werden und kann damit zur Unterstützung der Traceability beitragen. Zusätzlich besteht bei diesem Einsatz des Simulation Request weiterhin die Notwendigkeit auch bei der firmeninternen Verwendung eine klare Definition der Begriffe vorzusehen, da gegebenenfalls nicht ausschließlich Simulationsexperten beteiligt sind. Die Erkenntnis hinsichtlich Verwendung von Simulation und Modelling Request widerspricht damit jedoch nicht der Aussage, dass Simulation-Request und Modelling-Request die strukturierte und „standardisierte“ Zusammenarbeit zwischen Partnern erleichtern und unterstützen können.

Die gewonnenen Erkenntnisse aus den Workshops wurden neutralisiert aufbereitet an die Verantwortlichen im TP3 übergeben und sind direkt in den aktuell vorliegenden Stand des Simulation Request und Modeling Request eingeflossen.

Traceability Demonstrator

Im Rahmen des SET Level Projektes wurden mit mehreren Industrievertretern Einzelworkshops zur Vorstellung, dem Review und Erprobung des Traceability Demonstrators TRACY durchgeführt. Dabei waren neben, den direkten Projektpartnern auch teilweise Kollegen aus dem Fachbereich der jeweiligen Häuser beteiligt. Für diese Workshops wurden häufig konkrete Anwendungsbeispiele aus dem Projekt verwendet, in denen die jeweiligen Partner direkt involviert waren, um einerseits den Einstieg für nicht direkt am Projekt beteiligte Personen zu erleichtern und andererseits auch den konkreten Mehrwert des Demonstrators zu zeigen.

Die Erkenntnisse aus den Workshops lieferten einen wichtigen Beitrag zur späteren industriellen Anwendbarkeit von TRACY und konnten zum größten Teil noch während der Laufzeit des Projektes im Demonstrator implementiert werden bzw. wurden in die Roadmap bis zum fertigen Produkt mit aufgenommen. Zu einigen der bereits implementierten Funktionalitäten zählen dabei beispielweise die grafische Darstellung der Abhängigkeiten der einzelnen Artefakte oder die im Demonstrator implementierten standardisierten Austauschformate.

Darüber hinaus wurde mit einem Industriepartner eine konkrete Evaluation des Traceability Demonstrators im Unternehmenskontext durchgeführt. Hierzu wurden reale Unternehmensprozesse zur Simulation von ADAS-Funktionen untersucht und identifiziert an welcher Stelle der Einsatz des Demonstrators einen Mehrwert für die digitale Durchgängigkeit der Prozesse leisten und die Traceability der Arbeitsabläufe unterstützen kann. Basierend darauf wurden Use Cases für den Einsatz im Rahmen der Simulation eines Bremssystems erarbeitet und eine Demonstrator-Instanz für die Nutzung im Unternehmen aufgesetzt. In das Vorgehen wurde ebenfalls der im Projekt entwickelte Credible Simulation Process, der als Template in TRACY hinlegt ist, mit einbezogen. Bereits während der Entwicklung konnten zusammen mit dem Industriepartner wertvolle Informationen und Anforderungen an den Demonstrator gesammelt werden. Diese betreffen zum Beispiel die Verlinkung von Artefakten, den Arbeitsfluss im Tool oder die Durchführung von Analysen. Der zusammen mit dem

Industrievertreter aus SET Level final entwickelte PoC, wurde im Unternehmen im Rahmen einer Präsentation und Life-Demo einem größeren Publikum vorgestellt. Generell lässt sich aus dieser industriellen Erprobung ableiten, dass sich der toolgestützte Credible Simulation Process gut auf interne Unternehmensprozesse mappen lässt und die Arbeitsabläufe vereinfachen kann und dass der Einsatz von nichtproprietären Systemen und die Verwendung von neutralen Standards für die einfache Integration und Anwendung in Industrieunternehmen (z.B. durch die Anbindung an GitLab) einen deutlichen Vorteil mit sich bringt. Weiterhin hat sich gezeigt, dass die toolgestützte durch den Credible Simulation Process geleitete, strukturierte Vorgehensweise Querbeziehungen und Anhängigkeiten leichter identifizierbar macht und es zudem ermöglicht implizites Wissen im Unternehmen explizit und nachverfolgbar zu machen.

Während der Projektlaufzeit wurden kontinuierlich Anforderungen an TRACY gesammelt, die sich aus der konkreten Anwendung im Projekt und aus dem dazu gegebenen Feedback ergeben. Die gewonnenen Erkenntnisse sind in die kontinuierlich laufende Weiterentwicklung des Demonstrators eingeflossen.

Zusätzlich wurden neben der allen Beteiligten im Projekt zugänglichen Instanz des Traceability Demonstrators, TRACY, mit der die Workshops durchgeführt wurden, mehrere Cloud-Instanzen aufgesetzt, um den Industriepartnern eine Möglichkeit zu bieten, eigene Daten, die nicht für alle im Projekt sichtbar sein sollen in den Demonstrator einzupflegen und damit die industrielle Anwendbarkeit und Erprobbarkeit der Software zu erleichtern.

Simulation Use Case

Zusätzlich zur Erprobung der Kooperationsprozesse wurde für die industrielle Erprobung der SET Level Ergebnisse, der Aufbau des Simulation Use Case 2 und die dafür entwickelten Modelle, bei einem Industriepartner getestet. Dazu wurde ein Team aus verschiedenen Firmen gebildet, welche die jeweiligen Rollen innerhalb der Erprobung repräsentieren. Dieses Vorgehen hat das Ziel eine möglichst realitätsnahe Erprobung des Simulation Use Case durchzuführen, da davon auszugehen ist, dass auch im „realen“ industriellen Umfeld eine Aufgabenverteilung stattfinden wird. Abbildung 107 zeigt schematisch das Vorgehen innerhalb des Projektes zur Erprobung des Simulation Use Cases.

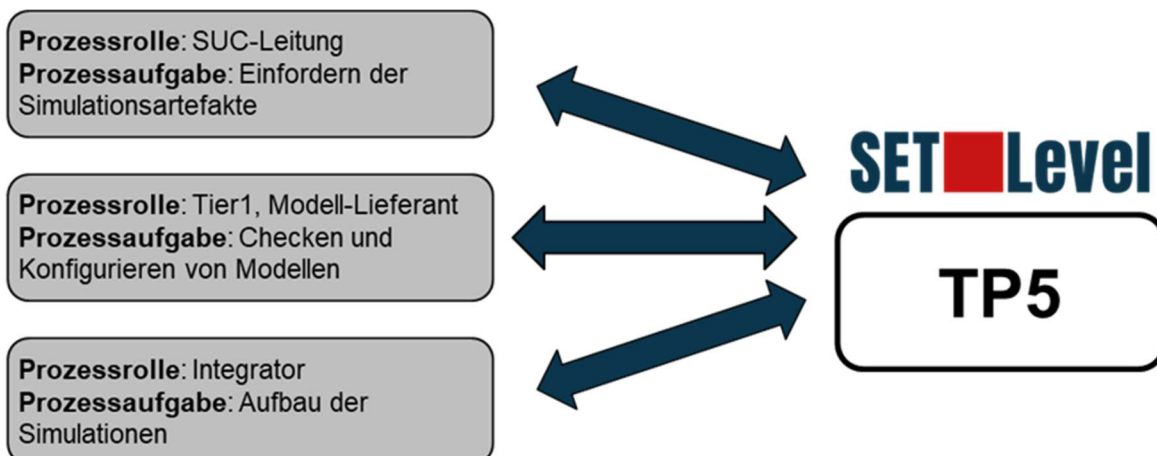


Abbildung 107: Schematische Darstellung der Simulation Use Case Erprobung

In der durchgeführten Erprobung konnte der Simulation Use Case 2 erfolgreich mittels drei verschiedener Architekturen umgesetzt werden: Innerhalb der Tools CarMaker (IPG) und ASM (dSPACE), monolithisch, sowie modular im Konzernumfeld.

Einige Erkenntnisse, die aus der Umsetzung des Simulation Use Cases 2 abgeleitet werden können und die mit großer Wahrscheinlichkeit generell auf den Aufbau und die Durchführung von Simulationen für das hochautomatisierte Fahren übertragbar sind, sind dass insbesondere für den Aufbau der Simulations-Infrastruktur und die Anpassung an existierende Simulationsframeworks das Vorhandensein von standardisierten Schnittstellen den Aufwand, der damit einhergeht, entscheidend verringern kann.

Darüber hinaus stellte auch die große Anzahl unterschiedlicher Stakeholder, Systeme, Lieferanten und Schnittstellen eine Herausforderung dar. Diese Situation stellt den industriellen Alltag jedoch gut dar. Durch die Anwendung in einem industriellen Umfeld wurde noch einmal deutlich, dass die praktische Umsetzung von Gesamtsimulationen für das hochautomatisierte Fahren eine große Herausforderung an die Modularisierung und Standardisierung der Simulation-Elemente darstellt.

Weiterhin wurde deutlich, dass die Integrationsfähigkeit der Modelle in die verwendeten Systeme, sowie deren Quantifizierbarkeit und Genauigkeit vor Beginn einer Simulationsaufgabe definiert sein sollte. Zudem sollten geeignete Prozesse vorhanden sein, um die unterschiedlichen Handlungsfelder geordnet abarbeiten zu können. Eine Möglichkeit hierzu besteht darin nach dem Credible Simulation Process (CSP) und dem Credible Modelling Process (CMP) vorzugehen. Mit diesen kann ein konsistenter Aufbau und verkürzter Prozessdurchlauf erreicht werden.

Die Ergebnisse der Erprobung des Simulation Use Cases wurden seitens des Industriepartners in einem Projektbericht zusammengefasst und eine um die Konzerninterna „gekürzte“ Version, in der nur Inhalte enthalten sind, die an das Gesamtprojekt gehen können, auf GitLab abgelegt und damit dem Projekt zur Verfügung gestellt.

Zusätzlich zur Erprobung des Simulation Use Cases fand ebenfalls ein Review des Traceability Demonstrators TRACY innerhalb des gleichen Projektes statt. Die Ergebnisse der TRACY-Erprobung sind bereits während der Erprobung und auch in Form eines finalen Reviews an die Verantwortlichen im TP4 übergeben worden. Hierbei wurde insbesondere die im Tool implementierte Projektplanung und die unterstützenden Funktionen zum Projektmanagement positiv hervorgehoben. Ein wesentliches Ergebnis des Reviews ist, dass das vorimplementierte Template des Credible Simulation Process dessen Anwendung und Befüllung deutlich erleichtert. Darüber hinaus wurden auch Verbesserungsvorschläge hinsichtlich Benutzerfreundlichkeit und Rollenmanagement erarbeitet.

Baselining im Projekt

Das SET Level Projekt hat sich für die Nutzung einer projekt- und partnerübergreifenden Datenablage entschieden. Im Rahmen des SET Level Projektes wurde dafür die Verwendung von GitLab forciert. Die Entscheidung wurde dabei aus Gründen der Datensicherheit zugunsten von git mit dem Server von GitLab getroffen. Dazu hat das DLR einen eigenen Server mit bundesdeutschem Daten Repository GitLab aufgesetzt und zur Verfügung gestellt.

Im Zuge des Projektes wurden die Nutzungsmöglichkeiten von GitLab dahingehend ausgebaut und im Projekt etabliert, dass das Repository nicht nur als persistente Datenablage verwendet werden kann, sondern auch die partner- und teilprojektübergreifende Zusammenarbeit unterstützt.

Die im git abgelegten Ergebnisse, Zwischenergebnisse und Arbeitsstände können von allen beteiligten Partnern eingesehen werden und sind diesen über die Downloadfunktion von GitLab zur Verfügung gestellt. Dies ermöglicht darüber hinaus, über die Nutzung von GitLab-Issues einen partnerübergreifenden Review von Ergebnissen und Arbeitsständen. Dieser kann dabei sowohl zentral (z.B. innerhalb einer Arbeitsgruppe) wie dezentral (durch Anmerkungen nicht direkt am Ergebnis beteiligter Partner) durchgeführt werden. Insbesondere im Zuge der Coronapandemie konnte hierdurch die Zusammenarbeit innerhalb des Projektes unterstützt werden. Beispielsweise wurde im Teilprojekt 3 ein teilprojektübergreifender Review des Credible Simulation Process über den Mechanismus der GitLab-Issues durchgeführt.

Durch die persistente Ablage von Zwischenständen, Projektergebnissen und die Verlinkung dieser Artefakte untereinander in git, konnte durch die Nutzung von GitLab zusätzlich das Baselining im Projekt unterstützt werden. Dies wurde speziell auch für die Entwicklungsstände der Simulation Use Cases angewendet. Auch hier besteht über die Verwendung von GitLab-Issues wieder die Möglichkeit konkretes Feedback zu geben bzw. einzelnen Reviews durchzuführen. Abbildung 108 zeigt exemplarisch die Möglichkeit des Baselining durch GitLab für den Simulation Use Case 1.

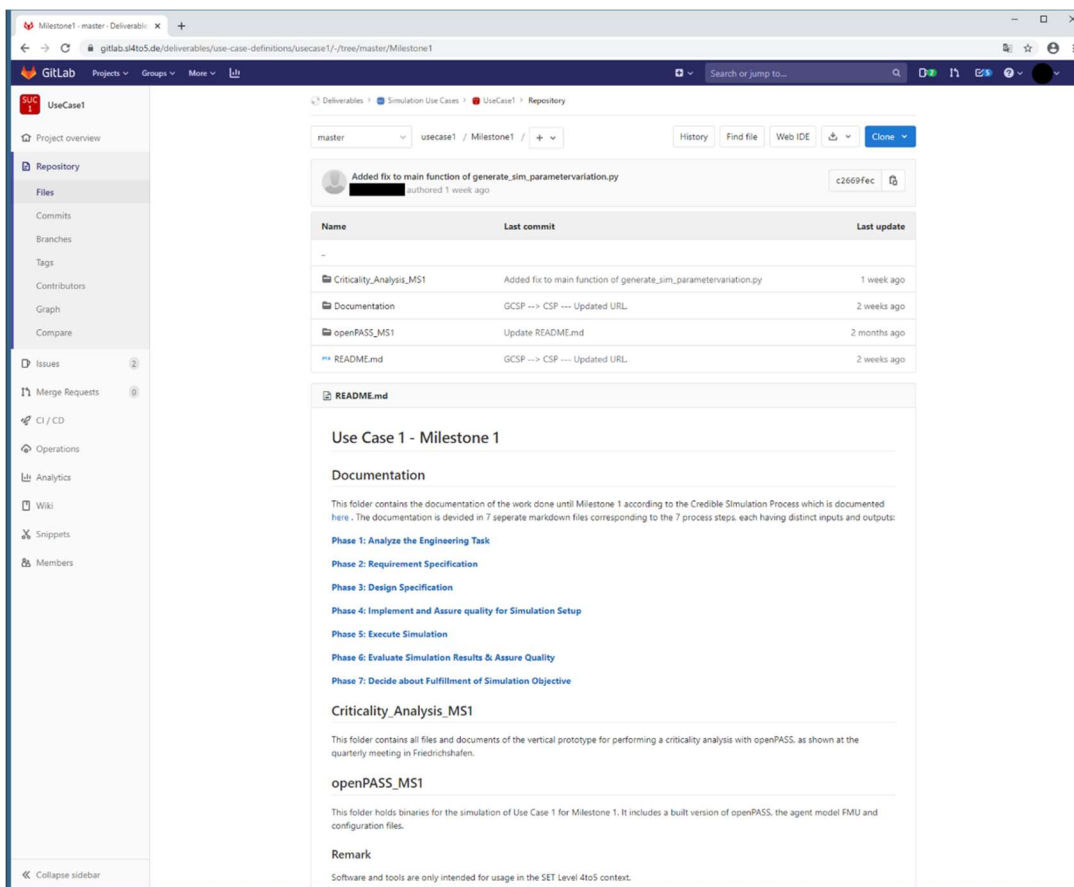


Abbildung 108: Baselineing am Beispiel des Simulation Use Case 1 für den ersten Meilenstein

Zusätzlich wurde der im Projekt entwickelte Traceability Demonstrator TRACY verwendet, um die im GitLab dokumentierten Inhalte der Simulation Use Cases nach dem Credible Simulation Process toolgestützt konsistent abzubilden. Dazu wurden die dokumentierten Inhalte in den standardisierten Formaten zip und XML aus TRACY ausgeleitet und ebenfalls im GitLab zu den jeweiligen Use Cases abgelegt.

Austausch mit anderen Projekten

Im Rahmen des Austausches mit anderen (Förder-)Projekten und Institutionen wurde von BMW der Austausch von SET Level Themen, speziell Themen mit Standardisierungscharakter und Themen mit Referenzcharakter in gemeinsamen Workshops mit SIP-adus durchgeführt. Darüber hinaus fand zusammen mit BMW, Bosch, dem DLR und PROSTEP eine Zusammenarbeit mit der prostep ivip Projektgruppe SmartSE (Smart Systems Engineering) und den ASAM Gremien statt. Insbesondere im Rahmen des im TP3 entwickelten Credible Simulation Prozess Framework und der Kooperationsprozesse fand dabei ein intensiver Austausch mit der prostep ivip Arbeitsgruppe SmartSE statt.

Nach Projektende wurden die im SET Level Projekt generierten Ergebnisse und Erkenntnisse im Themenkomplex Credible Simulation Prozess Framework an die SmartSE-Arbeitsgruppe übergeben, um die Arbeiten weiterzuführen und dabei neue Use Cases der Zusammenarbeit zu bearbeiten, die Ergebnisse zu versteigern und auch einen Personenkreis außerhalb der Automobilbranche zu adressieren, da in SmartSE auch Partner aus weiteren Branchen vertreten sind.

Zuvor wurde bereits zusammen mit Partnern aus dem TP3, TP4 und aus der SmartSE-Gruppe im Juni 2022 ein Workshop auf dem prostep ivip Symposium in Stuttgart durchgeführt. Einerseits war das Ziel die Vorstellung der Ergebnisse in Hinblick auf eine prozessgestützte, kollaborative Zusammenarbeit unter Einbeziehung von dafür definierten Standards und des im Projekt entwickelten Traceability Demonstrators TRACY. Andererseits lieferte der Workshop und die

dabei entstandenen Diskussionen, insbesondere mit Partnern nicht in SET Level mitwirken, wertvolle Inputs für die finalen Arbeiten im Projekt und auch im Hinblick auf das Abschlussexvent.

Darüber hinaus wurde unter Führung von BMW zusammen mit PROSTEP und dem DLR die Zusammenarbeit und der Austausch mit dem Schwesterprojekt VV-Methoden über die Projektlaufzeit stetig vorangetrieben.

1.2 Zahlenmäßiger Nachweis

Der zahlenmäßige Nachweis ist dem Erfolgskontrollbericht zu entnehmen.

1.3 Notwendigkeit und Angemessenheit der geleisteten Arbeit

Die Förderung ermöglichte eine umfassende, technisch tiefe Zusammenarbeit des Konsortiums, in dem alle relevanten Rollen von Akteuren im Sektor vertreten waren (OEM, Zulieferer, Technologieentwickler, Wissenschaft und Forschung). So konnten Fortschritte auf dem Feld der Simulation für den Test und die Absicherung automatisierter Straßenfahrzeuge erzielt werden, die eine breite Akzeptanz zu finden versprechen.

Insbesondere im Bereich der Standardisierung der Formate für Testbeschreibung, die simulationsinternen Schnittstellen und der darüber ausgetauschten Inhalte wurden in Abstimmung mit allen Beteiligten große Fortschritte erzielt. Auch die Nutzbarkeit der Setzungen für die praktische Anwendung konnte in gemeinsam umgesetzten Demonstrationen gezeigt werden. So hat der Ansatz, die akteurübergreifende Zusammenarbeit in einem Verbundprojekt zu fördern, wesentlich dazu beigetragen, im vorwettbewerblichen Bereich Übereinkünfte zu treffen, die zukünftig in wirtschaftliche Umsetzungen einfließen und fortschrittliche Technologien auf dem aktuellen Stand der Wissenschaft und Technik für alle Akteure verfügbar machen.

Durch die Förderung des Projekts SET Level wurde die Qualität der erreichten Ergebnisse insbesondere im Bereich der standardisierten Datenformate und der eingesetzten Tools durch die Abstimmung und Zusammenarbeit im Gegenstandsbereich der beteiligten Firmen und Institutionen in erheblichem Maße gesteigert. Ebenso wurden durch die Zusammenarbeit im Projekt umfangreichere Projektergebnisse erzielt, da die Zuwendung eine umfassendere Betrachtung und erweiterte Evaluation der Ergebnisse ermöglicht hat.

Für die Projektbearbeitung war spezifisches Fachwissen aus verschiedenen Bereichen notwendig, welches im Projektkontext von einzelnen Partnern jeweils nicht oder nur schwerlich umfassend abgedeckt werden konnte. Dies konnte durch die Zusammenarbeit im SET Level Projekt erreicht werden. Mit dem Anreizeffekt einer Förderung wurde eine solche Zusammenarbeit auf breiter Basis zwischen den Unternehmen und den Forschungsinstituten realisierbar. In diesem kooperativen Forschungsvorhaben konnte im Verbund eine valide Lösung für die adressierten SET Level Aufgabenstellungen demonstriert werden.

Zusammenfassend kann gesagt werden, dass die geleisteten Arbeiten und die damit verbundenen Aufwände, gemessen am erzielten Erkenntnisgewinn und Erfolg in einem angemessenen Verhältnis stehen.

Die Projektergebnisse werden von den Projektpartnern entsprechend ihrer Interessen weiterverfolgt.

1.4 Nutzen und Verwertbarkeit

Im Kontext des hoch automatisierten Fahrens treffen wir auf ähnlich komplexe Systeme und ähnlich hohe Ansprüche an Qualität und Zuverlässigkeit wie im Bereich der Luftfahrt. Daher ist es folgerichtig, sich an den entsprechenden Normen und Vorgehensweisen zu orientieren.

Zum Nachweis der Sicherheit und anderer Qualitäten eines hoch automatisierten Fahrzeugs ist es unabdingbar, seine exakte Konfiguration dokumentiert zu haben, die Zusammenstellung und Konfiguration der zugehörigen Entwicklungsartefakte zu managen (ISO 10007) und jederzeit den Nachweis führen zu können, wie jedes einzelne Artefakt zu Stande gekommen ist. Letzteres versteht man unter dem Begriff „Traceability“. Diese Traceability ist auch essenzielle Grundlage für die Glaubwürdigkeit von Entwicklungsergebnissen. Ohne die Unterstützung eines leistungsfähigen Software-Werkzeugs ist es praktisch unmöglich, die Vielzahl an Artefakten, Teilprozessen und ihrer wechselseitigen Abhängigkeiten unter Kontrolle zu halten. PROSTEP sieht daher ein großes Marktpotential für den Traceability-Demonstrator TRACY. Jedes Unternehmen, das autonome oder andere hochkomplexe Systeme am Markt anbieten will, wird

zukünftig ein Werkzeug mit dieser Funktionalität benötigen. Die Vermarktung wird über den PROSTEP-eigenen Vertrieb, aber auch über Vertriebspartner erfolgen. Eine weltweite Vermarktung, insbesondere über die PROSTEP-Tochtergesellschaft in den USA und PROSTEP-Vertriebspartnern, ist angestrebt. Und dies nicht nur im Umfeld des Projektes, der Automobilhersteller wie auch deren Zulieferer, sondern ebenfalls im Schiffbau, Aerospace, Energy, Anlagenbau etc. Auch hier gibt es einen hohen Bedarf nach einer solchen Lösung. PROSTEP strebt an, nach Projektende TRACY als Bestandteil des Produktes OpenCLM am Markt anzubieten. PROSTEP plant nach Projektende die im Projekt gesammelten Erfahrungen und den im Projekt erarbeiteten neuen Stand der Technik zur Ergänzung des (Strategie-) Beratungsangebots von PROSTEP heranzuziehen und auch so zur Verbreitung der Projektergebnisse beizutragen. Durch die enge Zusammenarbeit z.B. mit dem prostep ivip Verein, dessen Mitglieder zu 80% aus der Industrie und zu 20% aus der Forschung kommen, sowie dessen Vernetzung mit internationalen Standardisierungsgremien wie zum Beispiel ISO oder OMG, wird PROSTEP aktiv an der Standardisierung der Projektergebnisse mitwirken, und damit auch mittelbar die Forschungslandschaft prägen. Somit sieht PROSTEP für sich einen hohen wirtschaftlichen und wissenschaftlichen Nutzen des Vorhabens.

1.5 Bekannt gewordener Fortschritt

Während des SET Level-Projektes sind keine Fortschritte anderer Vorhaben auf dem Gebiet von SET Level bekannt geworden, die einen signifikanten Einfluss auf SET Level hatten.

1.6 Veröffentlichungen

1.6.1 Erfolgte Veröffentlichungen

Im Zuge der Ergebnisverbreitung erfolgte die Präsentation der SET Level-Umfänge wie in Tabelle 1 aufgelistet.

Tabelle 1: Erfolgte Veröffentlichungen von PROSTEP in SET Level

Nr.	Datum	Veranstaltung / Veröffentlichung	Ort	Titel	Autoren
1.	2021	https://setlevel.de/assets/forschungsergebnisse/Credible-Simulation-Process-v1.0.pdf .	Online	Credible simulation process.	Heinkel, H.-M., Steinkirchner, K., and the SET Level project
2.	Sep. 2020	prostep ivip Symposium 2020	Stuttgart	Realizing traceability for safety and certainty.	Heinkel, H.-M. and Vettermann, Dr. S.
3.	26.10.21	Proceedings of the 2021 NAFEMS World Congress	Online	Integrative simulation architecture for highly automated vehicles	Fischer, F. and Dr. C. Franke
4.	22.11.22	20. VDI-Kongress mit Fachausstellung – SIMVEC 2022 – Aktuelle Herausforderungen bei der Auslegung und Absicherung von Fahrzeugsystemen, Volume 2407 of VDI Nachrichten / VDI-Berichte,	Baden-Baden	Integrative Architektur für die Simulation hoch automatisierter Fahrzeuge	Franke, Dr. C.
5.	05.11.20	6th Symposium Driving Simulation	Stuttgart	Realizing traceability for safety and certainty.	Heinkel, H.-M. and Vettermann, Dr. S.
6.	01.09.21	Deutscher Luft und Raumfahrtkongress 2021	Online	Anwendung des Artefakt-Modells in einem generischen Simulationsprozesses	Bühler, C., Brandstätter M.
7.	08.11.21	Workshop der prostep ivip Projektgruppe SmartSE 2021	Darmstadt	Marktstand "TRACY"	Vettermann, S.
8.	26.11.21	öffentliches Web-Seminar des prostep ivip Vereins	Online	SmartSE: Traceable simulation results in a heterogeneous environment	D. Brück, H. Esen, H.-M. Heinkel, P. Lobner, P. Mai, S. Rude, and K. Steinkirchner

9.	08.06.22	prostep ivip Symposium 2022	Stuttgart	Trust and Traceability in Cooperative Simulation-Based Engineering	H.-M. Heinkel, P. Mai, P. Lobner, D. Brück, A. Wagner, S. Rude, M. Geissen
----	----------	-----------------------------	-----------	--	--

2 Abbildungsverzeichnis

Abbildung 1: Projekte der PEGASUS-Familie.....	5
Abbildung 2: Zusammensetzung der Partner von SET Level.....	6
Abbildung 3: Schnelle Iterationen in Projektmeilensteinen.....	6
Abbildung 4: Aufbau der Teilprojekte von SET Level.....	7
<i>Abbildung 5: Zeitplanung mit Meilenstein- und Abschlussberichten.....</i>	<i>8</i>
Abbildung 6: Anforderungs- und Reviewprozess	10
Abbildung 7: Weiterentwickelter Anforderungs- und Reviewprozess	11
Abbildung 8: Referenzszenario RS202 im Verkehrsraum VR202	12
Abbildung 9: 6-Layer Modell [2] als Basis für die Szenarienbeschreibung	13
Abbildung 10: SysML-Modell für OpenSCENARIO.....	14
Abbildung 11: Radardiagramm der für SET Level relevanten Standards, Erläuterungen zu Quadranten	15
Abbildung 12: Beispiel für das Radardiagramm der für SET Level relevanten Standards, eingeordnet nach Relevanz und Reifegrad.....	16
Abbildung 13: Exemplarische Verknüpfung der drei Architekturebenen mit dem Credible Simulation Prozess.....	17
Abbildung 14: Übertragung von Architektur nach SSP und weiter in andere Tools.....	18
Abbildung 15: Übersicht SUC1 Dokumentation in GitLab	19
Abbildung 16: Phase 1 SUC1 Dokumentation in GitLab	20
Abbildung 17: Modellanforderungen SUC1 in TRACY.....	21
Abbildung 18: Fußgängermodell als Artefakt in TRACY	22
Abbildung 19: Aufbau des Credible Simulation Process	23
Abbildung 20: Aktivitätsdiagramm des Credible Simulation Process	24
Abbildung 21: Feedback des Credible Simulation Process über GitLab Issues	25
Abbildung 22: Exemplarische Top-Level-Verknüpfung verschiedener Architekturebenen mit den Phasen des Credible Simulation Process	26
Abbildung 23: Aufbau des Credible Modeling Process	26
Abbildung 24: Zusammenhang zwischen Simulation-based Decision Process, Credible Simulation und Credible Modelling Process	27
Abbildung 25: Auszug aus der Checkliste des Credible Simulation Process	28
Abbildung 26: Anlagemöglichkeiten der Templates für Credible Simulation und Credible Modeling Process in TRACY (oben) und instanziiertes Template für den Credible Simulation Process in TRACY mit Subschritten (unten).....	29
Abbildung 27: Zusammenwirken der „Credible“-Prozesse mit dem prostep SmartSE Referenzprozess [1] inklusive Top-Level-Sicht der „Agreementphase“.....	30
Abbildung 28: Ausschnitt aus der Konsolidierung der Ergebnisse zum Kooperationsprozess (Aus Vertraulichkeitsgründen weichgezeichnet)	31
Abbildung 29: Simulation Request im Zusammenspiel mit dem Credible Simulation Prozess	31

Abbildung 30: Konsistenz von Prozess- und Datenstruktur	32
Abbildung 31: Übersicht über die Darstellung und Editierbarkeit von Metadaten im Traceability Demonstrator TRACY (Persönliche Daten geschwärzt)	34
Abbildung 32: Schnittstellen-Änderungsprozess (dargestellte Phasengrößen entsprechen nicht dem real nötigen Zeitaufwand)	35
Abbildung 33: Ablage der Traffic Spaces in GitLab.....	37
Abbildung 34: SUC und Szenarien verknüpft.....	37
Abbildung 35: Traffic Space Repository.....	38
Abbildung 36: Trace-Links in TRACY	38
Abbildung 37: Welche Schnittstellen?.....	40
Abbildung 38: Beispiele für Software im Bereich ADAS-Simulation	40
Abbildung 39: Schematische Darstellung der Ebenen der Unternehmensarchitektur im Kontext Simulation	41
Abbildung 40: Architektur-Dreibein	42
Abbildung 41: zur Frage 1 – Nachhaltige Passung von Architektur-Sichten	43
Abbildung 42: RFLP-ausgerichtetes V-Modell [11]	44
Abbildung 43: Erster Entwurf der Orientierungsmatrix.....	45
Abbildung 44: Endstand der Orientierungsmatrix	46
Abbildung 45: Funktionale Architektur eines realen Fahrzeugs (Fig. 26 des SaFAD-Papers [13]).....	48
Abbildung 46: Orientierungsmatrix der wichtigsten Architektur-Komponenten, in UML dargestellt	49
Abbildung 47: Geschäftsobjekte der Simulation	50
Abbildung 48: Primäre, sekundäre und tertiäre Simulationsergebnisse	50
Abbildung 49: AEB-relevante Komponenten aus SaFAD [13].....	51
Abbildung 50: Oberste Ebene der AEB-Simulationsarchitektur	52
Abbildung 51: Reales und simuliertes Universum.....	53
Abbildung 52: Funktionale Architektur des vollständigen Simulationssystems.....	54
Abbildung 53: Beide Teil-Ebenen der funktionalen Architektur samt Simulation Core.....	55
Abbildung 54: Beide Teil-Ebenen der funktionalen Architektur des Traffic Participants	56
Abbildung 55: Aggregation von vier Funktionen zu einer logischen Komponente	57
Abbildung 56: Konfigurationsgraph für den OSI-konformen Traffic Participant	58
Abbildung 57: Konfigurierbarer OSI-konformer Traffic Participant	58
Abbildung 58: Konfigurierbare OSI-konforme Komponente Perceptions and Behavior.....	59
Abbildung 59: Ausdifferenzierung der logischen Architektur	60
Abbildung 60: SaFAD-Paper [13] im Kontext der integrativen Architektur der Simulation	61
Abbildung 61: Zusammenspiel integrative Architektur und CSP	63
Abbildung 62: CSP-konforme Ableitung konkreter Architekturen aus abstrakteren Architekturen	64
Abbildung 63: Organisatorische Umsetzung der 3-Ebenen-Architektur	66

Abbildung 64: Ableitung von Modell-Rümpfen aus SSP-Dateien	66
Abbildung 65: Erster Versuch zur Übertragung von Architektur nach SSP und weiter in andere Tools	67
Abbildung 66: Darstellung des Powertrain-Blocks aus „Vehicledynamic and Actuators Model Detail 2“	67
Abbildung 67: Graphische Darstellung des SSP zum oben abgebildeten Powertrain-Blocks.	68
Abbildung 68: Konkrete SUC1-MS3 Architektur, Gesamtsicht	68
Abbildung 69: SUC1-MS3 Architektur, Sicht aufs Ego Vehicle	69
Abbildung 70: SUC1-MS3 Architektur, Sicht aufs Ego Vehicle, in easySSP importiert	69
Abbildung 71: Beschreibung eines Paares aus KEY und VALUE in SRMD	73
Abbildung 72: SRMD-Einträge zum selben KEY, aber mit unterschiedlichem TITLE	73
Abbildung 73: Verwendung von OpenStreetMap als externes Attributierungsschema	73
Abbildung 74: Realität – Simulationsmodel – Synthetisches Szenario	74
Abbildung 75: Metadaten des Szenarios TS2-4_RS-2-4	74
Abbildung 76: Inhaltsverzeichnis der ISO 10007 [12]	75
Abbildung 77: Konfigurationsmanagementbereiche gemäß ISO 10007	76
Abbildung 78: Schematische Darstellung der Artefakt-Relationen über den CSP hinweg	77
Abbildung 79: Gliederung der TRACY-GUI	78
Abbildung 80: Graphische Darstellung von Artefakten und ihren Relationen (Filter aktiviert).	79
Abbildung 81: Exemplarische Anwendung des grafischen Filtermechanismus	80
Abbildung 82: Auswahl der möglichen vorkonfigurierten Analysen	80
Abbildung 83: Exemplarische Durchführung der Impacted-Entities-Analyse	81
Abbildung 84: Exemplarische Darstellung zum Ableiten einer Baseline	81
Abbildung 85: Planned Baseline im Navigator (Tree View links) und Detail View (rechts) mit ausgeführter Ad-Hoc Baseline	82
Abbildung 86: Ansicht eines Configuration Items in der Detail View (rechts) mit Auswahlmöglichkeit der Version	83
Abbildung 87: Darstellung eines Composite Configuration Items in der Detail View (rechts) .	83
Abbildung 88: Löschen, Schließen, Öffnen und Exportieren von Projekten	84
Abbildung 89: TRACY OSLC Connector, exemplarisches Einbinden von Requirements	85
Abbildung 90: TRACY Issue Management Prozess	86
Abbildung 91: Top-Level-Sicht des Referenzprozesses zum „Proof of Concept“	88
Abbildung 92: Prüfung einzelner Projektergebnisse	89
Abbildung 93: Exemplarische Beschreibung eines konkreten Prozessschrittes	89
Abbildung 94: Prüfung von Projektergebnissen in deren Zusammenwirken	90
Abbildung 95: Beispielhafte Darstellung des Reviewtemplates	90
Abbildung 96: Dokumentation des Simulation Use Case 1 im Traceability Demonstrator anhand des Credible Simulation Process (ausgewählter Subschritt)	92
Abbildung 97: Der generische Einführungsprozess für SET Level-Artefakte	93
Abbildung 98: Vereinfachte Darstellung des generischen Einführungsprozesses	94

Abbildung 99: Vorgehen zur Ermittlung von Anforderungen die aus Industriesicht eine hohe Priorität haben.....	94
Abbildung 100: Exemplarische Ergebnisse eines Workshops (aus Vertraulichkeitsgründen unkenntlich gemacht)	95
Abbildung 101: Grobe Struktur Cameo.....	96
Abbildung 102: Ausleitung (aus Vertraulichkeitsgründen unkenntlich gemacht)	96
Abbildung 103: Aus Industriesicht priorisierte Anforderungen an automatisiertes Fahren, geordnet nach Themenclustern.....	97
Abbildung 104: Abbildung der Hauptphasen des Kooperationsprozesses in Anlehnung an [1], beteiligt Partner aus Vertraulichkeitsgründen nicht einzeln aufgelistet.....	98
Abbildung 105: Auszug aus den konsolidierten Ergebnissen der Workshops (aus Vertraulichkeitsgründen weichgezeichnet)	100
Abbildung 106: Zusammenhang zwischen Simulation / Modelling Request und Credible Simulation / Modelling Process.....	101
Abbildung 107: Schematische Darstellung der Simulation Use Case Erprobung.....	103
Abbildung 108: Baselineing am Beispiel des Simulation Use Case 1 für den ersten Meilenstein	105

3 Tabellenverzeichnis

Tabelle 1: Erfolgte Veröffentlichungen von PROSTEP in SET Level	109
---	-----

4 Literaturverzeichnis

- [1] SmartSE project group; *SmartSE Recommendation V2*; prostep ivip Association, 2018, abgerufen unter: https://www.prostep.org/fileadmin/downloads/prostep-ivip-Recommendation_PSI11_SmartSE_V2-0.zip am 23.03.2023
- [2] Scholtes Maike und Eckstein Lutz; *Systematic Categorization of Influencing Factors on Radar-Based Perception to Facilitate Complex Real-World Data Evaluation*; IEEE Intelligent Vehicles Symposium, Nagoya, 2021
- [3] SmartSE project group; *Simulation-based decision making and release*; White paper, prostep ivip Association, Germany, 2021
- [4] Eigner, M., Dickopf, T., and Apostolov, H.. The evolution of the V-model: From VDI 2206 to a system engineering based approach for developing cybertronic systems. In *14th IFIP International Conference on Product Lifecycle Management (PLM)*, pages 382–393, Seville, Spain. <https://hal.inria.fr/hal-01764166/document>., 2017.
- [5] OMG. Unified Modeling Language (UML), Object Management Group® (OMG®), Milford, MA, USA. <https://www.omg.org/spec/UML/About-UML/>, 2017.
- [6] Walden, D. D., Roedler, G. J., Forsberg, K. J., Hamelin, R. D., and Shortell, T. M., editors. *Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities*. Wiley, Hoboken, NJ, USA, 4th edition. INCOSE Systems Engineering Handbook, 2015.
- [7] International Council on Systems Engineering, <https://www.incose.org>
- [8] Systems Engineering Vision Working Group. INCOSE systems engineering vision 2020. Technical Product INCOSE-TP-2004-004-02, INCOSE, San Diego, CA, USA, 2007.
- [9] OMG. System Modeling Language (SysML), Object Management Group® (OMG®), Milford, MA, USA. <https://www.omg.org/spec/SysML/1.6/About-SysML/>, 2019.
- [10] Martin Eigner, Thomas Dickopf, Hristo Apostolov, P. Schaefer, KG. Faißt, and A. Keßler. System Lifecycle Management : Initial Approach for a Sustainable Product Development Process Based on Methods of Model Based Systems Engineering. In Shuichi Fukuda, Alain Bernard, Balan Gurumoorthy, and Abdelaziz Bouras, editors, *Product Lifecycle Management for a Global Market*, volume 442, pages 287–300, Springer , 2014.
- [11] Eigner, M., Dickopf, T. und Apostolov, H.. Interdisziplinäre Konstruktionsmethoden und -prozesse. *VDI e-Paper Reihe „Konstruktion“*, 09.11.2018.
- [12] <https://www.iso.org/standard/70400.html>
- [13] Wood, M., Wittmann, D., Liu, S., Knobel, C., Sandro Syguda, Wiltschko, T., Garbacik, N., O'Brien, M., Dannebaum, U., Weast, J., and Dorniede, B.. *Safety First for Automated Driving*. White paper, Aptiv, Audi, BMW, Baidu, Continental Teves, Daimler, FCA, HERE, Infineon Technologies, Intel, Volkswagen, Germany, April 2019.
- [14] Technical Report ISO/TR 4804:2020 - Road vehicles — Safety and cybersecurity for automated driving systems — Design, verification and validation.
- [15] ISO/AWI TS 5083 - Road vehicles — Safety for automated Driving systems — Design, verification and validation
- [16] https://en.wikipedia.org/wiki/ISO_10007, abgerufen am 23.03.2023

Berichtsblatt

1. ISBN oder ISSN	2. Berichtsart (Schlussbericht oder Veröffentlichung) Schlussbericht	
3. Titel Verbundvorhaben: SETLevel4to5 – Simulationsbasiertes Entwickeln und Testen von Level 4 und 5 Systemen Teilvorhaben: Unterstützung der Schnittstellenfunktion zw. Industrie und Forschungspartnern, Systems Engineering und Baselineing		
4. Autor(en) [Name(n), Vorname(n)] Dr. Christian Bühler, PROSTEP AG Dr. Carsten Franke, PROSTEP AG Dr. Martin Holland, PROSTEP AG	5. Abschlussdatum des Vorhabens 31.10.2022	
	6. Veröffentlichungsdatum	
	7. Form der Publikation Sonstiges	
8. Durchführende Institution(en) (Name, Adresse) PROSTEP AG Dolivostr. 11 64293 Darmstadt	9. Ber.-Nr. Durchführende Institution	
	10. Förderkennzeichen 19A19004H	
	11. Seitenzahl 115	
12. Fördernde Institution (Name, Adresse) Bundesministerium für Wirtschaft und Klimaschutz 11019 Berlin	13. Literaturangaben 16	
	14. Tabellen 1	
	15. Abbildungen 108	
16. DOI (Digital Object Identifier) Das o.g. Teilvorhaben wurde im Rahmen des Gesamtvorhabens durchgeführt.		
17. Vorgelegt bei (Titel, Ort, Datum)		
18. Kurzfassung Dieser Bericht stellt die von PROSTEP durchgeführten Arbeiten und Beiträge zum SET Level Projekt dar. Im Projekt wurde dabei die Anforderungserhebung für die Simulation sowie für die Szenarienräume und Referenzszenarien für den identifizierten Anwendungsfall der „komplexen urbanen Kreuzung“ unterstützt. Zusätzlich wurde eine Sammlung und übersichtlichen Darstellung von Standards, die im Projektkontext von Relevanz sind, durchgeführt. Zur Sicherstellung der Nachvollziehbarkeit und Traceability mit Fokus auf eine partnerübergreifende Zusammenarbeit wurde ein Prozessframework definiert und erstellt. Es umfasst unter anderem die beiden selbstähnlichen Prozesse, Credible Simulation und Credible Modelling Process, sowie den Vereinbarungsprozess und dass Simulation bzw. Modelling Request. Es wurde die Erstellung und Beschreibung der Metadatenbeschreibung für Modelle unterstützt. Des Weiteren wurde eine exemplarische integrative Simulationsarchitektur mit entwickelt und aufgebaut. Für die einheitliche Darstellung der entwickelten integrativen Simulationsarchitektur wurde ein UML-Modell der Architektur nach dem RFLP-Ansatz über die verschiedenen Architekturebenen in Enterprise Architect erstellt. Ein weiterer Arbeitsschwerpunkt war die Entwicklung und der Einsatz eines Traceability-Tools. Dieses wurde während der Projektlaufzeit den Partnern zur Nutzung zur Verfügung gestellt und entsprechend den Anforderungen aus dem Projekt kontinuierlich erweitert und optimiert. Zusätzlich wurde die Erstellung einer prozesssicheren Gesamtbeschreibung unterstützt, welche für die Dokumentation der Simulation Use Cases auf Basis des im Projekt entwickelten Credible Simulation Process verwendet werden kann. Darüber hinaus wurde die industrielle Erprobung der Projektergebnisse unterstützt. Dazu zählen unter anderem die Erhebung industrieller Anforderungen an hochautomatisierte Fahrzeuge und die industrielle Evaluation des im Projekt entwickelten Kooperationsprozesses und Simulation Request.		
19. Schlagwörter Automatisiertes Fahren, Simulation, Prozesse, Baselineing, Traceability, Simulationsarchitektur, Systems Engineering		
20. Verlag	21. Preis	

Nicht änderbare Endfassung mit der Kennung 1947877-7

Document control sheet

1. ISBN or ISSN	2. type of document (e.g. report, publication) Veröffentlichung (Publikation)	
3. title SET Level 4to5 Simulation-based Engineering and Testing of Level 4 and 5 Systems: Subproject: Support of the interface between industry and research partners, systems engineering and baselining		
4. author(s) (family name, first name(s)) Dr. Christian Bühler, PROSTEP AG Dr. Carsten Franke, PROSTEP AG Dr. Martin Holland, PROSTEP AG	5. end of project 31.10.2022	
	6. publication date	
	7. form of publication Sonstiges	
8. performing organization(s) name, address PROSTEP AG Dolivostr. 11 64293 Darmstadt	9. originators report no.	
	10. reference no. 19A19004H	
	11. no. of pages 115	
12. sponsoring agency (name, address) Federal Ministry for Economic Affairs and Climate Action 11019 Berlin	13. no. of references 16	
	14. no. of tables 1	
	15. no. of figures 108	
16. DOI (Digital Object Identifier) The sub-project mentioned above was carried out as part of the overall project.		
17. presented at (title, place, date)		
18. abstract This report presents the work and contributions to the SET Level project carried out by PROSTEP. In this context, requirements were gathered for the simulation as well as for the scenario spaces and reference scenarios for the identified use case of the "complex urban intersection" were supported. In addition, a collection and clear presentation of standards relevant in the project context was performed. To ensure traceability with a focus on cross-partner collaboration, a process framework was defined and created. It includes the two self-similar processes, Credible Simulation and Credible Modelling Process, as well as the Agreement Process and the Simulation and Modelling Request. The creation and description of the metadata description for models was supported. Furthermore, an exemplary integrative simulation architecture was co-developed and built. For the uniform representation of the developed integrative simulation architecture, a UML model of the architecture was created in Enterprise Architect according to the RFLP approach across the different architecture levels. Another focus of the work was the development and use of a traceability tool. This was made available to the partners for use during the project and continuously expanded and optimized according to the requirements of the project. In addition, the creation of a process-safe overall description was supported, which can be used for the documentation of the Simulation Use Cases based on the Credible Simulation Process developed in the project. In addition, the industrial evaluation of the project results was supported. This includes, among other things, the survey of industrial requirements for highly automated vehicles and the industrial evaluation of the Cooperation Process and Simulation Request developed in the project.		
19. keywords Automated Driving, Simulation, Processes, Baselining, Traceability, Simulation Architecture, Systems Engineering		
20. publisher	21. price	

Nicht änderbare Endfassung mit der Kennung 1947878-13