

**SIMULATIONS-  
FRAMEWORK  
»FERAL«**



## Die Herausforderung

In allen Bereichen des Lebens entstehen derzeit neue digitale Ökosysteme, welche mithilfe von Software und Daten bestehende Produkte und Prozesse verbinden.

Eine zentrale Herausforderung dabei ist die Integration von Komponenten und Systemen. Diese Integration findet immer häufiger dynamisch und zur Laufzeit statt. Gerade Systeme, die sicherheitsrelevante Entscheidungen auch automa-

tisiert treffen und von denen im Extremfall Menschenleben abhängen, müssen dann besonders abgesichert und getestet werden. Dazu zählen zum Beispiel Fahrzeugplattformen, die Fahrfunktionen dynamisch empfangen und integrieren können. Mit Industrie 4.0 steht die industrielle Produktion vor ähnlichen Problemstellungen, wenn es darum geht, neue Geräte und Prozesse dynamisch in eine bestehende Anlage zu integrieren.

## Typische Fragestellungen

Gemeinsam mit unseren Projektpartnern konnten wir in der Vergangenheit virtuelle Prototypen erfolgreich einsetzen, um klassische Fragen zu Architekturentscheidungen oder Systemdesign zu beantworten:

- Wie lassen sich sicherheitsrelevante Fahrzeugfunktionen mit einem Smartphone fernsteuern, ohne die Sicherheitseigenschaften zu verletzen?
- Ist die neue Fahrzeugplattform sicher vor Cyber-Angriffen?
- Welche Netzwerktechnologie bietet genug Bandbreite für meine Fahrzeugplattform/Industrieanlage?
- Wie lässt sich meine Industrie-4.0-Produktion automatisch planen?
- Welche Auswirkungen hat die Einführung eines neuen Robotertyps in meiner Industrieanlage?
- Wie kann man einen Software-Lockstep zur Steuerung von Industrieanlagen realisieren?
- Wie kann man zum bestehenden System offene Schnittstellen hinzufügen und diese absichern?
- Wird die neue Komponente in die bestehende Infrastruktur passen?
- Wurden alle Anforderungen in der Systemarchitektur berücksichtigt?

## Die Lösung

Die Lösung FERAL von Fraunhofer IESE ist eine Simulationsplattform mit unterschiedlichen Bausteinen. Diese ermöglichen es, auch komplexe, heterogene Szenarien in ein Prüfzenario zu integrieren und gezielt Eigenschaften im geschützten virtuellen Raum mithilfe digitaler Zwillinge zu überprüfen. Dies kann die korrekte Funktionsweise einer Fußgängererkennung sein oder das fehlerfreie Zusammenspiel zweier Funktionen von unterschiedlichen Herstellern.

## Die Vorteile virtueller Entwicklung

- Kosten- sowie Zeitersparnis bei strategischen Entwicklungen und Änderungen von Systemkonzepten (Steuergeschichte, Netzwerke etc.)
- Frühzeitige Erkennung von Fehlentwicklungen
- Absicherung von Entscheidungen mit messbaren Resultaten
- Berücksichtigung der Anforderungen verschiedener Stakeholder (Marketing, Entwickler, Nutzer, Projektleitung etc.)
- Fokus auf Lösungskonzepte statt auf Probleme
- Testen von Systemkonzepten im Kontext von Fehlern und Angriffen (bspw. Hackerangriffe)

## Anwendungsbeispiel Autonomes Fahren / Virtuelle Validierung

Autonomes Fahren stellt die Automobilbranche vor große Herausforderungen. Dazu gehören technische Herausforderungen, etwa »Wie erkennt das Fahrzeug zuverlässig Passanten, die die Fahrbahn kreuzen?« oder »Welche Entscheidung muss die Künstliche Intelligenz treffen?«. Die größte Herausforderung liegt in der funktionalen Sicherheit – es muss gewährleistet sein, dass das autonome Fahrzeug genauso sicher oder sicherer als ein Fahrzeug mit menschlichem Fahrer ist.

Aufgrund der erhöhten Komplexität eines autonomen Fahrzeugs stoßen dabei klassische Validierungstechniken schnell an ihre Grenzen. Experten des TÜV Süd<sup>1</sup> gehen davon aus, dass 100 Millionen kritische Situationen getestet werden müssen, um eine einzelne Fahrzeugfunktion ausreichend zu testen. Klassische Feldtests können die notwendige Abdeckung niemals erreichen, da eine Anzahl kritischer Situationen in echten Fahrsituationen zu selten auftritt. Aus diesem Grund geht BMW<sup>2</sup> davon aus, dass für autonome Fahrzeuge 95 % der Tests virtuell mithilfe von Simulationen durchgeführt werden müssen.

*“If a test driver were to drive 1,000 kilometers a day, more than 5,000 drivers would need to be in continuous operation 365 days a year to ensure the validation of a fully automated car.”  
VW inside*

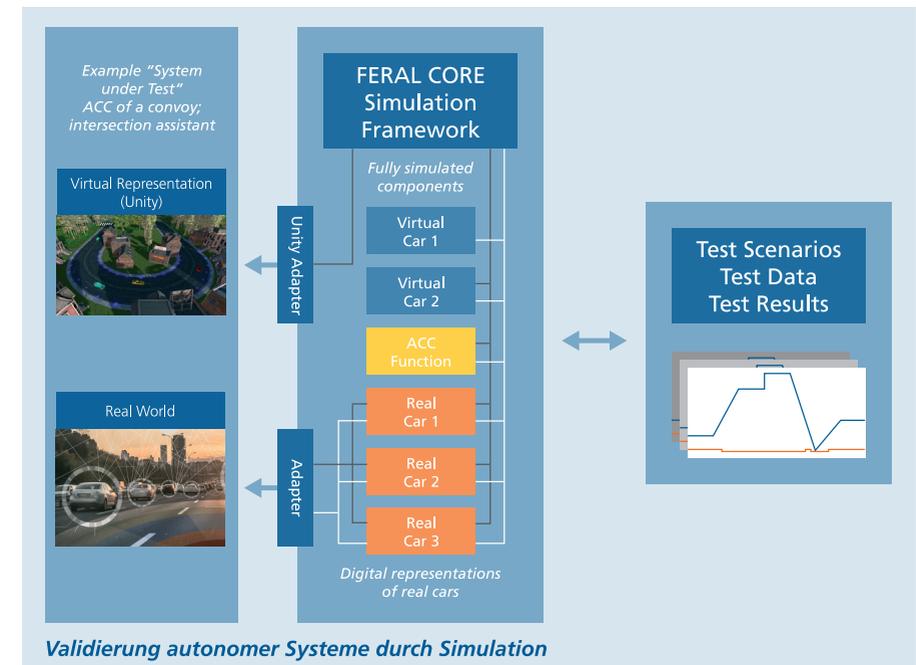
Der Vorteil: Simulationen können vorhandene Szenarien durch Variationen anpassen, sodass kritische Situationen gehäuft auftreten. Beispielhaft soll das Testen eines Kreuzungsassistenten betrachtet werden: In echten Fahrsituationen kommt es selten vor, dass zwei Fahrzeuge gleichzeitig an einer Kreuzung ankommen und das Assistenzsystem einen Notbremsvorgang durchführen muss. Entsprechend aufwändig wären Feldtests in realen Fahrsituationen.

Hier bietet die Simulation die Möglichkeit, virtuelle Fahrzeuge zu erzeugen, die die kritische Situation auslösen und somit die Validierung des Kreuzungssystems sicherstellen.

Bei der virtuellen Validierung autonomer Systeme genügt es jedoch nicht, nur die

Sensordatenverarbeitung und die Korrektheit der Künstlichen Intelligenz zu betrachten. Die verwendeten Prozessoren oder die Fähigkeit der in den Fahrzeugen eingesetzten Netzwerktechnologien können dabei einen Flaschenhals bilden, der eine rechtzeitige Reaktion verhindert. Aus diesem Grund kann die Validierung eines

autonomen Systems nur durch eine Simulation geschehen, welche das komplette Fahrzeug als digitalen Zwilling mit Sensoren, Steuergeräten und Aktuatoren sowie die sie verbindende Netzwerktechnik enthält. Dazu ist die Kopplung verschiedener Simulationsmodelle notwendig.



<sup>1</sup> <https://www.tuv-sud.com/industry/automotive-transportation/automotive-solutions/pegasus-project-autonomous-cars-ensuring-safety-on-the-road>

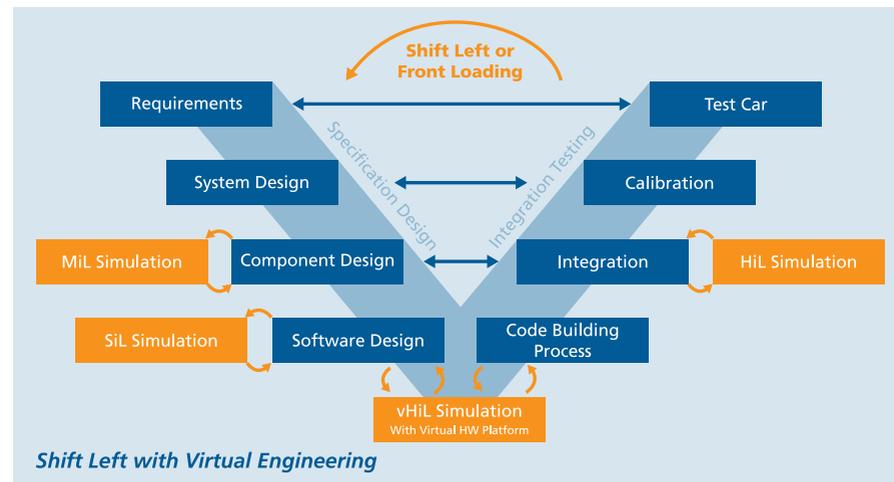
<sup>2</sup> <https://www.golem.de/news/zulassung-autonomer-autos-die-laengste-fahrpruefung-des-universums-1611-124139.html>

## Unsere Technologie

Das vom Fraunhofer IESE entwickelte FERAL-Simulationsframework erstellt virtuelle Prototypen durch die Kopplung von Simulationsmodellen und Simulatoren, existierendem Code und virtuellen Hardware-Plattformen. Dadurch können die Auswirkungen von Entscheidungen frühzeitig überprüft werden.

In einem sehr frühen Entwicklungsstadium kann die Simulation rein auf Modellen basieren; dann spricht man von Model-in-the-Loop-(MiL)-Simulation. FERAL nutzt dazu beispielsweise UML Zustandsautomaten oder Activity-Diagramme oder die Kopplung mit Matlab Simulink. In diesen frühen Phasen werden oftmals kritische Designentscheidungen getroffen, die durch diese Simulationen abgesichert werden können. Sind bereits erste Realisierungen von Software vorhanden, können diese in einer Software-in-the-Loop-(SiL)-Simula-

tion mit den existierenden Modellen kombiniert werden. Dadurch können genauere Vorhersagen über das Systemverhalten getroffen werden, oder es kann in einem Back-to-Back-Test die Compliance zu den zuvor erstellten Modellen überprüft werden. Parallel zum Design der Software wird im klassischen V-Modell das Design der Hardware vorangetrieben. Diesen Schritt unterstützt FERAL durch die Bereitstellung von virtuellen Hardwareplattformen, also Prozessor- und Netzwerkmodellen, auf welchen in einer virtuellen Hardware-in-the-Loop-(vHiL)-Simulation die Softwarekomponenten deployed werden können. All diese Simulationen (MiL, SiL und vHiL) dienen dazu, Fehler möglichst in frühen Entwicklungsphasen zu entdecken, wodurch sich die Anzahl von teuren Hardware-in-the-Loop-(HiL)-Simulationen und Integrationstests reduzieren lässt.



Semantic Models		Failure Modes	Host Platforms	Simulated Abstract Platforms
Event-Triggered	Agent Scheduler	Loss	MPI Clusters	1-Core Processor
Finite State Machines	Discrete Time	Corruption	Soft Real Time	2-Core Processor
Continuous Time	Service Domain	Delay	Single-Core	4-Core Processor
		Insertion	Windows	8-Core Processor
		Sequencing	Multicore	3-Core Aurix
		Masquerading	Linux	Tile 64 Processor
Behavior Models		Network Simulation Models		
Java Script	Activities	GPRS/UMTS	Ideal Wire	Flexray
Dataflow	SystemC	WiMax	Point To Point	CAN
C	Groovy	RS 232	TT Ethernet	Ethernet
FMU	State Machines	RS 485	LTE	Wireless
Sequences	Simulink	Ideal Bus	Ideal Wireless	Bluetooth
C++	Java			X86
				ARM

*Virtual Engineering Technology – Selection of FERAL Simulation Components*

## Unsere Dienstleistungen

Mit unserer Erfahrung unterstützen wir Sie gerne bei der Entwicklung neuer System- und Softwarekonzepte. Um Entscheidungen abzusichern, Systeme zu optimieren oder etwas ganz Neues zu kreieren, arbeiten wir gerne mit Ihren Ingenieuren zusammen.

Kunden, die virtuelle Prototypen selbst einsetzen möchten, unterstützen wir bei der Entwicklung angepasster Simulationslösungen sowie bei der Umsetzung eigener Konzepte zur Kopplung von Simulatoren. Haben Sie Fragen zur virtuellen Simulation mit FERAL? Sprechen Sie uns an!

## **Ansprechpartner**

Dr. Thomas Kuhn  
thomas.kuhn@iese.fraunhofer.de  
Telefon +49 631 6800-2177  
www.iese.fraunhofer.de

Ralf Kalmar  
ralf.kalmar@iese.fraunhofer.de  
Telefon: +49 631 6800-1603  
www.iese.fraunhofer.de

## **Fraunhofer-Institut für Experimentelles Software Engineering IESE**

Fraunhofer-Platz 1  
67663 Kaiserslautern

## **Das Fraunhofer IESE**

Seit mehr als 20 Jahren ist das Fraunhofer IESE in Kaiserslautern eine der weltweit führenden Forschungseinrichtungen auf dem Gebiet der Software- und Systementwicklungsmethoden. In mehr als 1.200 Projekten haben seine wissenschaftlichen Mitarbeiter bereits ihre Kompetenzen in den Bereichen Prozesse, Architektur, Security, Safety, Requirements Engineering und User Experience eingebracht. Das Institut beschäftigt sich mit innovativen Themen wie Industrie 4.0, Big Data und Cyber-Security, ist Technologie- und Innovationspartner für die digitale Transformation in den Bereichen Autonomous & Cyber-Physical Systems und Digital Services und erforscht das Zusammenspiel von eingebetteten Systemen und Informationssystemen in digitalen Ökosystemen.

Das Fraunhofer IESE ist eines von 72 Instituten und Einrichtungen der Fraunhofer-Gesellschaft. Zusammen gestalten sie die angewandte Forschung in Europa wesentlich mit und tragen zur internationalen Wettbewerbsfähigkeit Deutschlands bei.