

# **Modulare Softwarearchitektur für die interaktive Simulation von Maschinen- und Fahrzeugsystemen in virtuellen Umgebungen (SARTURIS)**

Prof. Dr.-Ing. habil. G. Kunze<sup>1</sup>, Dipl.-Ing. T. Penndorf<sup>2</sup>

Technische Universität Dresden, Institut für  
Fördertechnik, Baumaschinen und Logistik

01062 Dresden

## **Kurzfassung**

Ziel des Projektes ist die Entwicklung einer modularen Softwarearchitektur für die interaktive Simulation von Maschinen- und Fahrzeugsystemen in virtuellen Umgebungen. Die Simulationssoftware soll die ganzheitliche Beziehung zwischen Bediener, Maschine und Arbeitsprozess abbilden. Aus diesem Grund sind die erforderlichen numerischen Berechnungen interaktiv und in Echtzeit durchzuführen. Die dafür notwendigen Optimierungen sollen durch den Einsatz von Codegeneratoren erreicht werden.

Die ganzheitliche Abbildung des komplexen Systems erfordert die Zerlegung in Teilsysteme. Für die Simulation dieser Teilsysteme existieren eine Reihe von Programmsystemen, deren Kopplung in einem Echtzeitumfeld nicht möglich ist. Deshalb soll unter Berücksichtigung der bereits vorhandenen Informationen des Ingenieurdatenmanagements eine abstrakte Beschreibung des Systems Bediener-Maschine-Prozess erstellt und mittels Codegeneratoren in ein ganzheitliches Berechnungsmodell überführt werden.

Dieses Projekt wird gemeinsam mit den Projektpartnern ITI GmbH, cad scheffler GmbH und Dornier GmbH bearbeitet.

---

<sup>1</sup> kunze@ifbl.tu-dresden.de

<sup>2</sup> penndorf@ifbl.tu-dresden.de

# 1. Einleitung

## 1.1. Interaktive Simulation von Maschinen- und Fahrzeugsystemen

Auf der Basis der enormen Leistungsfähigkeit moderner Rechner können immer mehr Informationen über ein technisches System bereits in den frühen Phasen in dessen Entwicklung einfließen. Es ist möglich, komplexe Systemeigenschaften und Wechselwirkungen zu erkennen und frühzeitig bei der Systemgestaltung zu berücksichtigen.

Für Maschinenbauer bedeutet dies einen Trend zum Einsatz von Simulationswerkzeugen bereits während der Projektierung und Entwicklung neuer Produkte. Ganzheitliche Wirkungen auf ein technisches System können in „Virtuellen Labors“ untersucht werden. Der Einsatz von Simulationssystemen ist die logische Fortführung herkömmlich durchgeführter analytischer Berechnungen, welche aus Gründen der Komplexität, Genauigkeit und Übersichtlichkeit für die Lösung der heutigen Aufgaben nicht mehr ausreichend sind.

Die Anwendung der Simulationstechnologie bei der Entwicklung mobiler Maschinen und Fahrzeuge stellt zusätzliche Anforderungen an das Simulationssystem. Wesentlich für die Qualität der Simulationsergebnisse ist die Berücksichtigung des Bedienerinflusses auf das Systemverhalten der Maschine. Dieser ist nicht durch ein Modell zu ersetzen. Die Simulation ist deshalb interaktiv zu gestalten, d.h. alle Berechnungen müssen in Echtzeit erfolgen.

## 1.2. Stand der Technik

Die Spezifizierung des Begriffes Simulationssystem führt zu einer Unterteilung der Aufgabenstellung in eine *formale Beschreibung* des technischen Systems mit Modellen sowie deren *Berechnung*.

Die Formulierung der Modelle erfolgt gegenwärtig nach den vorgegebenen Regeln des verwendeten Simulationssystems. Bis auf einige Ansätze sind diese Regeln noch nicht standardisiert. Die vorhandenen Simulationssysteme lassen sich in zwei Kategorien unterteilen. Einerseits existiert eine Reihe von Systemen und Programmen, welche als monolithisch bezeichnet werden können. Auf der anderen Seite kommen mathematisch-technische Programmsysteme zum Einsatz.

Beispiele für monolithische Programme sind die Mehrkörpersysteme alaska, SimPack und MSC.visualNastran, welche Software-Tools zur Untersuchung des dynamischen Verhaltens komplexer mechanischer Systeme darstellen, sowie das Programm ITI-Sim und DRESP zur Simulation von Antriebssträngen.

Diese Programme verfügen meist über sehr übersichtliche Möglichkeiten der Modelleingabe unter Verwendung der Symbolik und Syntax von Ingenieuren. Sämtliche Routinen zur Modellbildung und Berechnung sind fester Bestandteil dieser Programme und bis auf ihre Para-

meter (z. B.: Schrittweite und Verfahrensordnung der numerischen Integration) nicht veränderlich.

Ein gravierender Mangel der vorhandenen Anwendungen sind unzureichende Schnittstellen für Kosimulation und Datenaustausch. Da diese Programme einzeln nicht in der Lage sind, die komplexe Aufgabenstellung zu lösen, ist nur über eine Kopplung der Teilsysteme die Möglichkeit einer Berechnung des Gesamtsystems gegeben. Aufgrund der beschriebenen Kapselung auch berechnungsinterner Daten, ist diese Kopplung nicht sinnvoll zu realisieren. Strukturierte Simulationsprogramme zur Abbildung des Gesamtkomplexes Maschine-Umgebung-Prozess sind gegenwärtig nicht verfügbar.

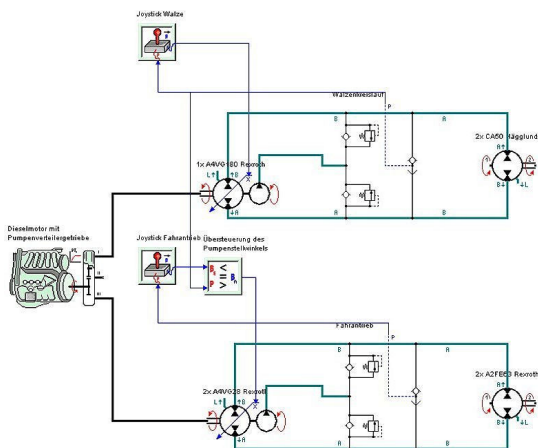


Abbildung 1: ITI-Sim

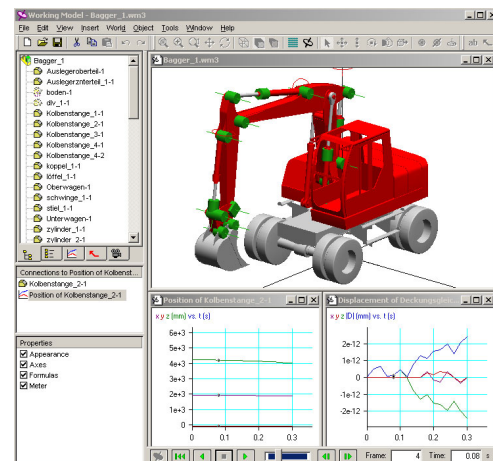


Abbildung 2: MSC.visualNastran

Vorhandene Software-Produkte zur Simulation komplexer Systeme, wie z.B. das System ITI SIM der ITI GmbH, stoßen zunehmend an Grenzen bei der Abbildung komplexer Maschinen- oder Fahrzeugsysteme. Vernetzte Systeme werden künftig in der Maschinenbau- und Fahrzeugindustrie eine zunehmend dominierende Rolle spielen. Die Beherrschung dieser ständig steigenden Komplexität erfordert die Einführung offener modularer Funktionsstrukturen, die eine bessere Wiedererkennung und Nachvollziehbarkeit ermöglichen. Erst durch die ganzheitliche Simulation der Teilsysteme einer Maschine oder eines Fahrzeuges wird diese Modularität beherrschbar.

Mathematisch-technische Software wie Matlab oder Maple ermöglicht dagegen die Modellbeschreibung in sehr abstrahierter Form. Dies erhöht die Flexibilität des Einsatzes, da jedes System berechnet werden kann, für welches eine mathematische Beschreibung vorliegt. Gleichzeitig ist mit dem Einsatz dieser Systeme jedoch ein erheblicher Verlust an Übersichtlichkeit verbunden, da durch die rein mathematische Beschreibungsform die Darstellung der ingenieurtechnischen Zusammenhänge verloren geht. Deshalb werden diese Programme nur bei der Bewältigung spezifischer Problemstellungen eingesetzt und stellen für viele Anwender

keine brauchbare Alternative dar. Die direkte Programmierung der modellbeschreibenden Gleichungen und Differentialgleichungen in einer Hochsprache führt meist auf unübersichtliche, schwer zu modifizierende und ineffiziente Quellcodes.

Die Analyse der bestehenden Möglichkeiten für die Simulation komplexer technischer Systeme unter dem Aspekt des "Virtual Prototyping" zeigt, dass gegenwärtig eine Reihe von Einschränkungen existieren, welche mit den vorhandenen Methoden eine erfolgreiche Abbildung eines Gesamtsystems verhindern.

### **1.3. Lösungsansatz**

Die angestrebte ganzheitliche Simulation soll das komplexe Verhalten des Systems *Maschine-Arbeitsprozess-Umgebung* abbilden. Vorhandene Softwaresysteme sind aufgrund der beschriebenen Einschränkungen dazu nicht in der Lage. Zur Lösung des Problems soll ein neuer Ansatz für eine Architektur von Simulationssoftware angewendet und überprüft werden. Die Schaffung der Softwarearchitektur gliedert sich in zwei Teile:

- neue Methoden und Entwicklungsumgebungen für die formale Beschreibung technischer Systeme mit Modellen sowie
- ein zu entwickelnder Code-Generator für spezialisierte, d.h. modelloptimierte Berechnungsalgorithmen, welche Echtzeitanforderungen erfüllen.

Es hat sich gezeigt, dass schon sehr begrenzte Modelle eine hohe Zahl an Informationen benötigen, welche bereits zum großen Teil in anderen Systemen zur Verfügung stehen. Geometrische Informationen, Trägheitsmomente, Schwerpunktlagen u.ä. Daten können beispielsweise direkt aus den Datenbasen verwendeter CAD-Modelle übernommen werden. Um nicht mit redundanten Daten arbeiten zu müssen, ergeben sich mit dem Einsatz von Simulationslösungen spezielle Anforderungen. Für die umfangreichen Simulationsdaten, welche ebenso wie die verschiedenen Varianten im Kontext mit verwendeten Verfahren und Simulationsläufen verwaltet werden müssen, sind herkömmliche PLM<sup>3</sup>-Komponenten nur bedingt ausgelegt. Deshalb ist die Kopplung des Softwaresystems mit ingenieurtechnischen Informations- und Arbeitswerkzeugen (CAD, PLM) ein wesentliches Ziel der Software-Umgebung. Diese Kopplung stellt eine wesentliche Erweiterung vorhandener Software-Systeme, wie sie u.a. von der CAD Scheffler GmbH entwickelt werden, dar.

---

<sup>3</sup> PLM-Product Lifecycle Management

## 1.4. Ziele und innovativer Gehalt

Im Mittelpunkt des Forschungsvorhabens stehen qualitätsverbessernde Methoden zur Erzeugung effizienter, robuster und flexibler Simulations-Softwaresysteme für die Maschinen- und Fahrzeug-Simulation.

Als Resultat wird die Möglichkeit gegeben, alle für die Berechnung relevanten Informationen einer komplexen Maschinenstruktur unabhängig von der Implementierung der Berechnungsprogramme zu verwalten. Diese Informationen repräsentieren eine zusätzliche Sicht auf die Produktdaten und bilden die Entwicklungszyklen während der Berechnungsphase ab. Die einheitliche abstrakte Beschreibung des technischen Systems erfolgt unabhängig von der Implementierung der Simulationssysteme.

Besonders für kleine und mittelständische Unternehmen soll ein Beitrag geschaffen werden, bestehende Produkte (Simulationssoftware, PLM-Lösungen) mit neuen Modulen zu erweitern und somit auf dem Markt mit innovativen Lösungen bestehen zu können, bzw. ihre Stellung auszubauen.

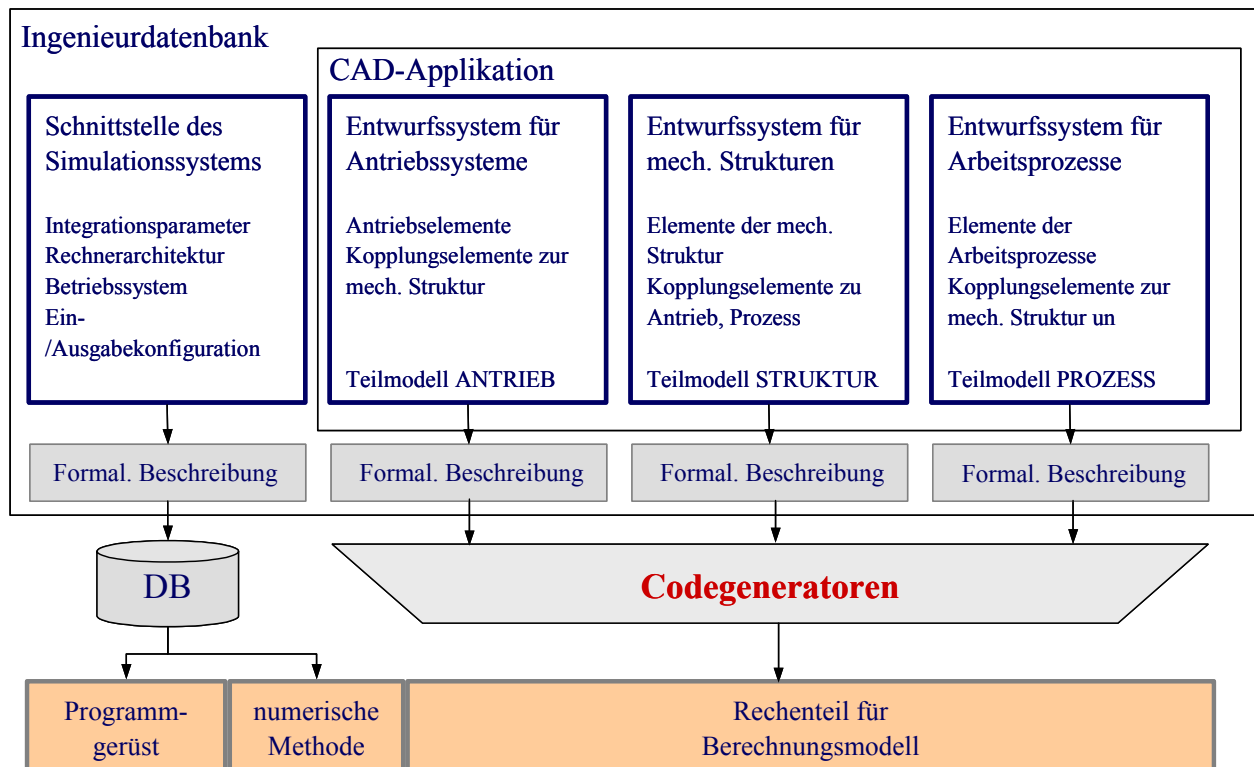


Abbildung 3: Modulare Softwarearchitektur

Bei der projektgemäßen Schaffung spezialisierter Objektmodelle mittels Codegeneratoren entstehen neuartige und grundlegende Methoden zur Durchführung von komplexen echtzeitfähigen Simulationsprozessen. Durch die vom Konstruktionsdatenmanagement gesteuerte Codeerzeugung entsteht ein modulares ganzheitliches und echtzeitfähiges Simulationsmodell.

Somit ist die interaktive Simulationstechnik für Maschinen- und Fahrzeughersteller rentabel zu betreiben. Die Entwicklung und Testung von Maschinenkonstruktionen mit vernetzten Teilsystemen (Embedded Systems) sind mit der neu erarbeiteten Simulationssoftware sehr gut durchführbar.

## 2. Projektstatus

### 2.1. Interaktiver Simulator für Arbeitsmaschinen

Die erworbenen Grundkenntnisse der Modellbildung und numerischen Berechnung wurden aktiv bei der Schaffung eines Forschungs- und Entwicklungssimulators angewandt und erweitert. Dieser interaktive Simulator für mobile Arbeitsmaschinen wurde zu Beginn des Jahres 2003 an der TU Dresden, Institut für Fördertechnik, Baumaschinen und Logistik, in Betrieb genommen. Er ermöglicht den Aufbau einer virtuellen Realität (VR) für verschiedene und komplexe Systemmerkmale in einer hochgradig live wahrnehmbaren Umgebung.

Der Simulator selbst stellt softwaretechnisch eine große Herausforderung dar, da verschiedene autonome Soft- und Hardware-Komponenten als geschlossenes System arbeiten.

Bei der Entwicklungsarbeit dieses Gerätes wurden eine Reihe von Unzulänglichkeiten herkömmlicher Simulationssoftware festgestellt, was die Erarbeitung eigenständiger Rechenprogramme erforderte. Aus den dabei gesammelten Erfahrungen ergibt sich der Lösungsansatz für das Forschungsprojekt. Die Forschungsergebnisse werden am Beispiel einer Referenzmaschine an diesem Simulator implementiert und verifiziert.

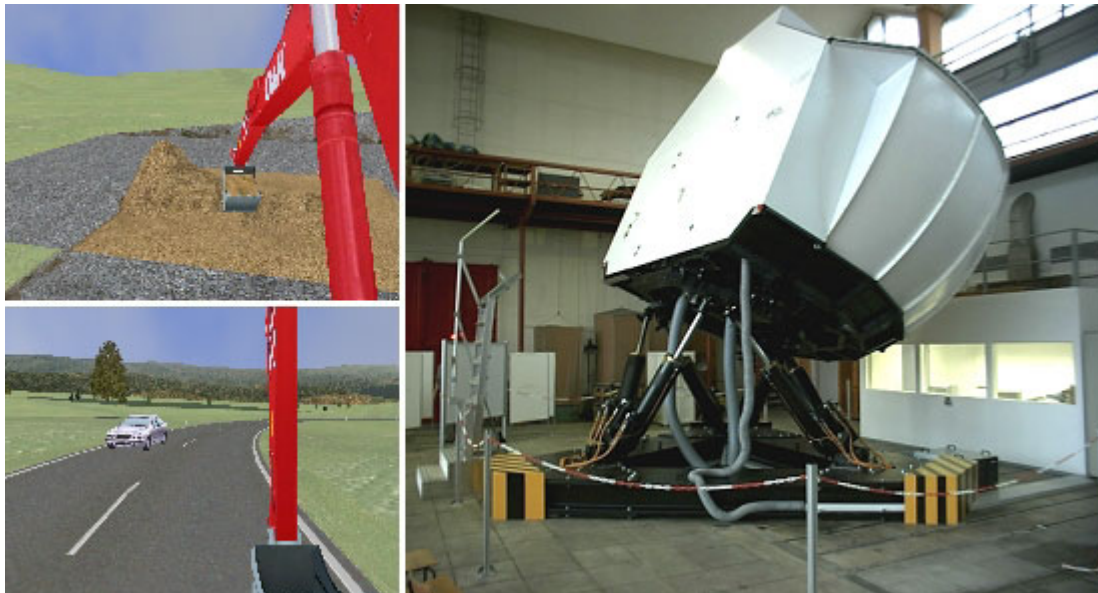


Abbildung 4: Interaktiver Simulator für Arbeitsmaschinen

## 2.2. Softwarekonzept

Das Konzept zur Realisierung der geplanten Softwarearchitektur sieht vor, dass zuerst eine geeignete Metasprache zu definieren ist, mit deren Hilfe komplexe technische Systeme beschrieben werden können. Diese Beschreibung wird den mathematischen Charakter der Maschinenkomponenten, d.h. ihre Übertragungs- und System-Funktionen widerspiegeln. Diese Funktionen sind im Allgemeinen Differential- und algebraische Gleichungen. Ein Objekt wird funktional ausschließlich über diese Gleichungen beschrieben, was die Erweiterbarkeit garantiert.

Die Sprache wird objektorientiert und durch eine XML ähnliche Syntax erweiter- und anpassbar sein. Zur Abbildung von Maschinenmodellen und Auswertung der Informationen können gängige Editoren eingesetzt werden. Weiterhin ist die Kopplung an technische Entwurfssysteme (CAD) vorgesehen.

Vorhandene Produktdatenmodelle werden ergänzt, um alle für die Berechnung notwendigen Informationen zu berücksichtigen. Dazu wird eine Überarbeitung und Ergänzung herkömmlicher Entity-Relationship-Modelle notwendig sein.

Der Übergang zu lauffähigem Simulationscode wird durch Codegeneratoren ermöglicht. Diese existieren für einige technische Spezialdisziplinen (Antriebssimulation) bereits in den eingesetzten „monolithischen“ Systemen. Es ist zu überprüfen, inwieweit vorhandene Generatoren für die Aufgabenlösung verwendet werden können, was wesentlich davon abhängt, ob sich das bestehende System in die skizzierte Prozesskette einbinden lässt.

Existiert ein solches System nicht (Starrkörpersystem), so sind die entsprechenden Codegeneratoren zu erstellen. Mittels solcher Lösungen wird die vorgestellte Beschreibungssprache gelesen und ein übersetzbarer Quellcode erzeugt, welcher das beschriebene technische Teilsystem mathematisch beschreibt und innerhalb der Berechnung abbildet. Die mathematische Beschreibung enthält alle Systemmatrizen, sowie deren Ableitungen bis zur Ordnung  $n$ . Diese Ordnung wird von der eingesetzten numerischen Methode festgelegt, deren Berechnungsvorschrift in einer Codedatenbank hinterlegt ist.

Die Datenbank enthält ebenfalls alle notwendigen Module, um ein Simulationsprogramm mit Ein-/Ausgabeschnittstellen, numerischer Methode und modellspezifischem Teil auf Anforderung zu erstellen und ablaufen zu lassen.

## 3. Arbeitsplan und Ausblick

Schwerpunkt in der ersten Projektphase ist die Definition eines Formates für die systemunabhängige Beschreibung der Simulationsmodelle. Dieses Format stellt die Schnittstelle zwischen Ingenieurwissen und Programm dar und ist somit die wichtigste Grundlage der Arbeiten. Ausgehend von einer Analyse der vorhandenen Vielfalt an Maschinenkomponenten und vorhandener Systemstandards ist eine allgemeingültige, objektorientierte Beschreibungssyntax als Basis der Modellbildung zu definieren. Dabei erfolgt soweit wie möglich eine Orien-

tierung an bestehenden Industriestandards (Modellica, XML). Die angestrebte Offenheit erlaubt die Verbindung zu bestehenden Entity-Relationship-Modellen des Produktdatenmanagements und somit eine Eingliederung der Simulationsmodelle in bestehende Informations- und Managementsysteme.

Im zweiten Arbeitspaket ist auf der Basis des definierten Metaformates ein Entwurfssystem für die ingenieurmäßige grafisch orientierte Aufbereitung der zu simulierenden Systeme und Abbildung der entsprechenden Modelle zu schaffen.

Es wird ein Prototyp für das Teilsystem „Mechanische Strukturen“ erarbeitet, welcher mit dem definierten Metaformat zu beschreiben ist. Er stellt auch die Grundlage für die Erarbeitung des Entwurfssystems dar. Im Ergebnis der Prototypenentwicklung für mechanische Strukturen werden Methoden für die Entwicklung der Systemsoftware für den Arbeitsprozess und das Antriebssystem untersucht und Lösungen für die Systemintegration erarbeitet.

Das Entwurfssystem soll eng mit bestehender Standardsoftware (CAD-Systeme) verbunden werden. Parameter sollen soweit wie möglich automatisch bestimmt werden. Die beschriebene Verbindung der Simulationsdaten mit dem Ingenieurdatenmanagement wird durch dieses Entwurfssystem hergestellt. Die Parameter haben eine für die Qualität der Simulation enorm wichtige Bedeutung. In einigen Fällen können diese aus vorhandenen Informationen erzeugt werden (Masseparameter aus CAD, Kennlinien aus Kataloginformationen). Die dafür notwendigen Software-Werkzeuge sind zu entwickeln.

Die eigentlichen Berechnungsmodelle sollen aus den im Entwurfssystem erzeugten Metainformationen unter Beachtung der im Entwurfssystem verwendeten Parameter erzeugt werden. Die physikalisch-technischen Eigenschaften des Systems sind in diesem Modell nicht mehr identifizierbar. Dafür stellt das Berechnungsmodell die abgebildeten mathematischen Systemzusammenhänge (Systemfunktionen, Jacobi-Matrizen) über standardisierte Schnittstellen zur Verfügung. Die der Codeerzeugung vorausgehenden symbolischen Operationen sollen dabei soweit wie möglich durch standardisierte Bibliotheken (Maple, NTL, Maxima) ausgeführt werden. Das Berechnungsmodell lässt sich nach bisherigem Kenntnisstand in drei Bereiche unterteilen. Dies ist

- ein Programmgerüst für die Initialisierung und Einordnung in die (Echtzeit-) Betriebssystemumgebung, den Start der Berechnung, die Gewährleistung der Interaktion, das Sammeln und Weiterleiten der Ergebnisse
- einen numerischen Teil, spezifiziert durch die numerische Methode und Routinen zur Anwendung dieser Methode auf ein Modell
- den modellspezifischen Teil, welcher alle von der numerischen Methode benötigten Größen berechnet.

Die ersten beiden Teile können aufgrund fest definierter Spezifikationen in Hochsprache (C) bereitgestellt werden.

Dies ermöglicht eine prinzipielle Erweiterbarkeit sowohl auf andere Rechnerarchitekturen und Betriebssysteme als auch hinsichtlich der Anwendung verschiedener numerischer Methoden. Der dritte Teil des Berechnungsmodells ist der Teil, welcher speziell für jedes Simulationsmodell zur Verfügung gestellt werden muss. Dieser wird entsprechend den vorhandenen Informationen erzeugt. Er stellt höchste Anforderungen an Effizienz. Unter Beachtung der Flexibilität wird ein Hochsprachen Quelltext erzeugt, welcher durch Standardcompiler dem entsprechenden alternativen Rechnersystem zugänglich gemacht wird.

Während der Simulation sollen die aktuell berechneten Ergebnisse ständig ausgegeben und mitgeschrieben werden können. Dafür sind Schnittstellen- und Auswertungs-Werkzeuge zu entwickeln, welche die notwendigen Informationen in erforderlicher Form darstellen. Eine Anwendung der Ergebnisse für die Steuerung interaktiver Simulatoren wird umgesetzt. Die dafür entwickelten Standards (HLA<sup>4</sup>) werden bei der Entwicklung der Ausgabemechanismen berücksichtigt. Bei der Programmierung der Auswertungs-Software steht die Kopplung mit PLM als Basis der Integration in die Infrastruktur der Unternehmen im Mittelpunkt.

Es ist ein wesentlicher Anspruch des Forschungsprojektes, dass die Softwaremodule als Grundlage einer breiten Nutzung in der Industrie verwendet werden können. Deshalb wird besonderes Augenmerk auf die Schaffung eines globalen Anwendungskonzeptes gelegt, welches die Nutzung der entstandenen Software bei Maschinen- und Fahrzeugherstellern als Simulationswerkzeuge in der Produktentwicklung (in Ergänzung zu eingesetzten CAD- und PLM-Applikationen) ermöglicht. Die Dornier GmbH wird die Forschungsergebnisse als Pilot-Anwender unmittelbar bei der Entwicklung neuer Simulatoren einsetzen.

---

<sup>4</sup> HLA – Standard einer allgemeinen Architektur für verteilte Simulator-Software (Quelle: IEEE Standard 1516, September 2000)