

Entwicklungsprozesse und -werkzeuge

Ausarbeitung für die Einführungsveranstaltung

zur Projektgruppe Autolab

am

Fachbereich 4 (Informatik) an

der Universität Dortmund

im

Wintersemester 2007 / 2008

Der zugehörige Vortrag wurde am 20. Oktober 2007 in Wuppertal gehalten.

von

Boris Nikolai Konrad

boris.konrad@uni-dortmund.de

8. Fachsemester

Inhaltsverzeichnis

1. Entwicklungsprozesse
 - 1.1 Unterstützungsprozesse
 - 1.2 Kernprozess
2. Werkzeuge
3. Literatur

1. Entwicklungsprozesse

1.1 Unterstützungsprozesse

Bei der Betrachtung der Entwicklung von elektronischen Systemen sind neben dem eigentlichen Entwicklungsprozess (Kernprozess) eine Reihe von Unterstützungsprozessen relevant (siehe Abbildung 1). Diese sind weitgehend unabhängig von der Softwareentwicklung und können daher für Entwicklungsprojekte jedweder Art meist in gleicher oder ähnlicher Art und Weise eingesetzt werden.

Konfigurationsmanagement

Jedes Produkt hat einen Lebenszyklus, welchen man üblicherweise in die drei Phasen Entwicklung, Produktion und Betrieb/Service unterteilt. Dabei können für einzelne Komponenten diese Zyklen unterschiedlich lang ausfallen, da etwa die Lebensdauer eines Fahrzeugs im Normalfall länger ist als die Lebensdauer einzelner Komponenten, die nach einiger Zeit zu ersetzen sind. Auch die Entwicklungsphase des Gesamtfahrzeugs kann die Entwicklungszeit einzelner elektronischer Bauteile, die einem viel schnelleren technologischen Fortschritt unterliegen, deutlich überschreiten.

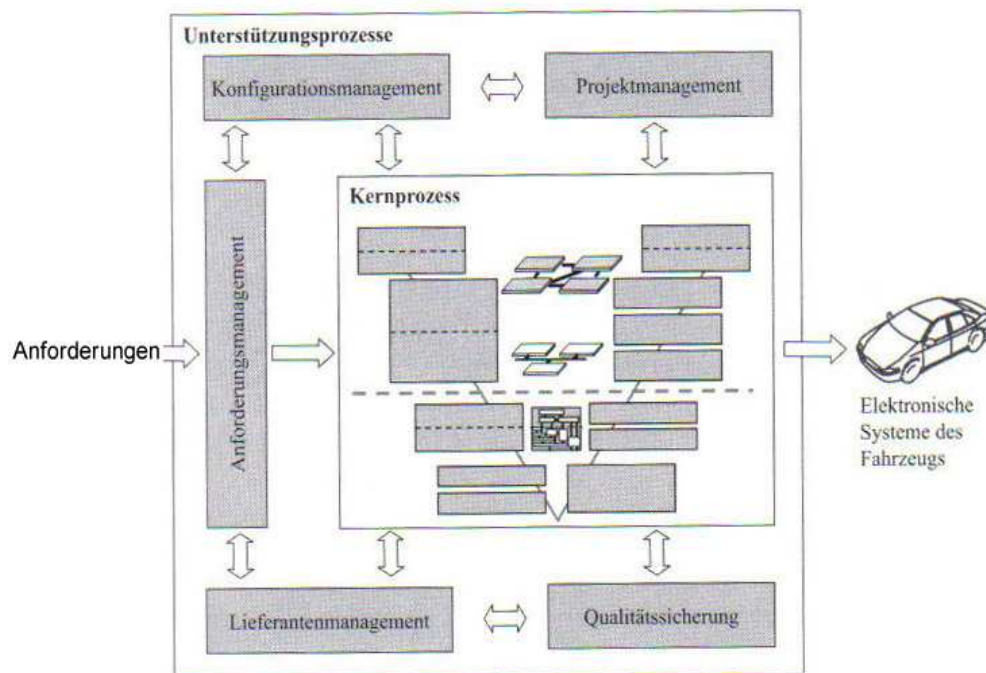


Abbildung 1: Unterstützungsprozesse¹

Für die Entwicklung ist es von Bedeutung, die unterschiedlichen Anforderungen etwa an die Steuergeräte während der drei verschiedenen Phasen zu kennen und zu unterscheiden. Oft ist etwa unterschiedliche Software-Funktionalität gefordert. Gleichzeitig ist beim Entwurf des Fahrzeugs und der eingebauten Elektronik zu beachten, dass die Kunden gewisse Anforderungen an die Erweiterbarkeit des Fahrzeugs stellen. So kann es nötig sein, verschiedene Varianten der gleichen Komponente zu erstellen oder Schnittstellen anzubieten, die bei Auslieferung nicht genutzt werden, etwa wenn die Möglichkeit bestehen soll ein Radio nachzurüsten.

Projektmanagement

Die Entwicklung eines Fahrzeugs erfüllt alle wesentlichen Merkmale eines Projektes wie etwa eindeutige Aufgabenstellung, definierbare Ziele und zeitliche Befristung. Daher ist eine Projektplanung gemäß den Regeln des Projektmanagements sinnvoll.

¹ [Schäuffele 2006]

Zunächst sind die Projektziele zu unterscheiden. Dies sind Terminziele, Kostenziele, sowie Qualitätsziele. Für alle drei Arten sollte also eine eigene Zielplanung erfolgen.

Wesentliche Elemente des Projektmanagements sind dabei die Zerlegung in Teilprobleme, deren Finalisierung mit dem Erreichen definierter Meilensteine erfolgt. Hierbei sollte eine umfassende Planung und Verfolgung der benötigten und verfügbaren Ressourcen erfolgen, um die Kostenziele des Gesamtprojektes stets zu gewährleisten. Auch ist insbesondere bei großen Projekten wie dem Entwurf eines Fahrzeugs eine Terminplanung und die terminliche Koordination verschiedener beteiligter Projektgruppen notwendig. Hierzu erfolgt auch eine Rollenvergabe an Mitarbeiter und Teams gemäß ihrer Qualifikationen und Aufgabenbereiche.

Anforderungsmanagement

Im Optimalfall beginnt der Entwicklungsprozess mit dem Erfassen aller Anforderungen an das zu entwickelnde Produkt. Hierbei werden (meist sprachlich ausformuliert) sämtliche Anforderungen erfasst, kategorisiert und verknüpft. Dabei sind alle Benutzer des Produkts, bei einem Automobil also mindestens Fahrer, Beifahrer, Servicetechniker, andere Verkehrsteilnehmer und Gesetzgeber, zu berücksichtigen. Hierbei ist darauf zu achten, dass die Anforderungen nicht bereits versteckte Vorgaben an die Realisierung beinhalten. Beispielsweise wäre die Aussage „Der Tankinhalt soll in Litern anstatt mit der Skala $\frac{1}{4}$ - $\frac{1}{2}$ - $\frac{3}{4}$ - 1 angezeigt werden“ bereits ein Lösungsvorschlag für die eigentliche Anforderung „Die Reichweite des Fahrzeugs soll genauer angezeigt werden“.²

Neben der anfänglichen Erfassung der Anforderungen gehört auch deren weitere Verfolgung zum Anforderungsmanagement, um sicherzustellen, dass alle getroffenen Entwicklungsentscheidungen den Anforderungen genügen. Hierzu ist es zwingend erforderlich, dass im Anforderungsmanagement auch eine Verknüpfung zur logischen und technischen Systemarchitektur vorgesehen ist.

² [Schäuffele 2006, Seite 137]

Lieferantenmanagement

Da häufig eine Vielzahl an Subsystemen durch Lieferanten erstellt wird, ist beim eigentlichen Hersteller des Produkts ein effizientes Lieferantenmanagement notwendig, da er allein die Verantwortung für die Funktion der angelieferten Komponenten im Gesamtsystem trägt. Hierzu ist insbesondere eine genaue Definition aller Schnittstellen notwendig, um dem Lieferanten klare Vorgaben zu machen, wie seine Komponente eingebunden werden soll und welche Verbindung zum Restsystem bestehen oder auch nicht bestehen wird. Hierbei sind neben rein technischen Gegebenheiten auch organisatorische Problemstellungen wie rechtliche Problematiken oder Konkurrenzsituationen verschiedener Lieferanten untereinander zu bedenken.

Man unterscheidet bei der zwischen Spezifikations- Schnittstellendefinierung (linker Ast im V-Modell, siehe Abbildung 2) und Integrationsschnittstellen (rechter Ast).

Qualitätssicherung

Die Qualitätssicherung steht im engen Zusammenhang mit dem Anforderungsmanagement und Projektmanagement und soll sicherstellen, dass alle Anforderungen sowie alle Kosten- und Qualitätsziele erfüllt werden. Unter die Qualitätssicherung fallen Richtlinien zur Qualitätssicherung, worunter vorbeugende Maßnahmen fallen wie beispielsweise der Einsatz geeigneter Entwicklungsprozesse und Werkzeuge. Ebenso gehören Maßnahmen der Qualitätsprüfung zur QS. Hierbei ist insbesondere das ausführliche Testen zur frühzeitigen und sicheren Entdeckung von Fehlern zu nennen. Dies schlägt sich auch im V-Modell (siehe Abbildung 2) nieder, in welchem nach jedem Integrationsschritt ein Testschritt vorgesehen ist.

Allgemein unterschieden werden Verifikation und Validation. Zur Verifikation gehören der Komponententest, der Integrationstest und der Systemtest, während der Akzeptanztest bei dem das System gegen die Benutzeranforderungen getestet wird, zu den Validationsmaßnahmen gehört.

1.2 Kernprozess

Es gibt verschiedene Modelle, welche den Entwicklungsprozess von Systemen beschreiben. Ein in der Automobilindustrie sehr verbreitetes und auch auf die Entwicklung elektronischer Systeme anwendbares Modell ist das V-Modell (siehe Abbildung 2).

Unterschieden wird zwischen zwei Sichtweisen auf das System: Die zunächst zu entwerfende logische Systemarchitektur stellt eine abstrakte Sicht auf das Funktionsnetzwerk her, während in der technischen Systemarchitektur die konkrete Realisierung in Steuergeräten des Fahrzeugs dargestellt wird.³

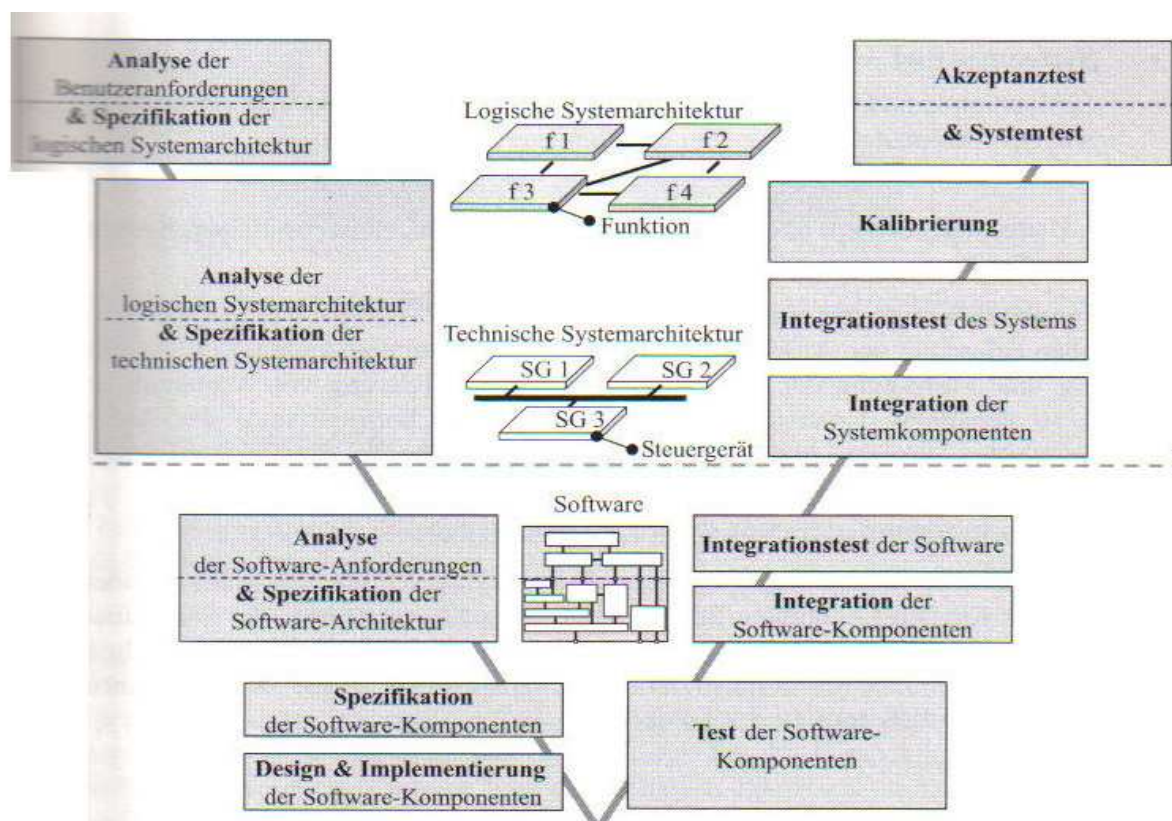


Abbildung 2: Kernprozess⁴

Das V-Modell bietet den großen Vorteil, einen durchgängigen Entwicklungsprozess zu modellieren, der deshalb notwendig ist, weil so die Wechselwirkungen zwischen der

³ [Schäuffele 2006, Seite 16]

⁴ [Schäuffele 2006, Seite 23]

Entwicklung von Software, Elektronik und des Gesamtsystems am besten eingebunden werden. Außerdem sind hierbei die Maßnahmen der Qualitätssicherung bereits mit eingebunden.

Im so dargestellten Kernprozess zur Entwicklung von elektronischen Systemen und Software werden folgende neun Entwicklungsprozesse unterschieden:

- Analyse der Benutzeranforderungen und Spezifikation der logischen Systemarchitektur
- Analyse der logischen Systemarchitektur Spezifikation der technischen Systemarchitektur
- Analyse der Software-Anforderungen und Spezifikation der Software-Architektur
- Spezifikation der Software-Komponenten
- Design, Implementierung und Test der Software-Komponenten
- Integration der Software-Komponenten und Integrationstest der Software
- Integration der Systemkomponenten und Integrationstest des Systems
- Kalibrierung
- Systemtest und Akzeptanztest

2. Werkzeuge

Telelogic DOORS

Telelogic DOORS ist eine Software für Anforderungsmanagement und seit 1992 auf dem Markt. Mit inzwischen über 250.000 Anwendern ist sie in ihrem Gebiet Marktführer. Mit dem Programm ist es möglich, Anforderungen der verschiedenen Benutzer sowie gesetzliche Vorgaben und Standards effizient zu erfassen, zu verknüpfen und zu analysieren. So können auch konkrete Softwareanforderungen mit ausformulierten Benutzeranforderungen in Zusammenhang gebracht werden. Hierdurch wird insbesondere die Nachverfolgbarkeit während des Entwicklungsprozess verbessert und die interne Kommunikation und Zusammenarbeit aller am Projekt beteiligten erhöht.

Matlab

Matlab ist eine in Industrie und Forschung weit verbreitete Software zur Lösung mathematischer Probleme und zur grafischen Darstellung der Ergebnisse. Im Gegensatz zu Computer-Algebra-Systemen (wie z.B. Mathematica) ist Matlab vor allem für numerische Simulationen und Datenanalyse gedacht. Hersteller der kommerziellen Software ist die Firma MathWorks.

In Matlab wird eine eigene proprietäre Programmiersprache benutzt, die auf dem jeweils benutzten Rechner interpretiert wird. Für Matlab gibt es eine ganze Reihe von Erweiterungen, sowohl von MathWorks selbst, als auch von Drittanbietern. Dies sind zum einen ganze Softwarepakete die auf Matlab aufbauen und zum anderen umfangreiche Bibliotheken beispielsweise für die Signalverarbeitung oder Statistik.

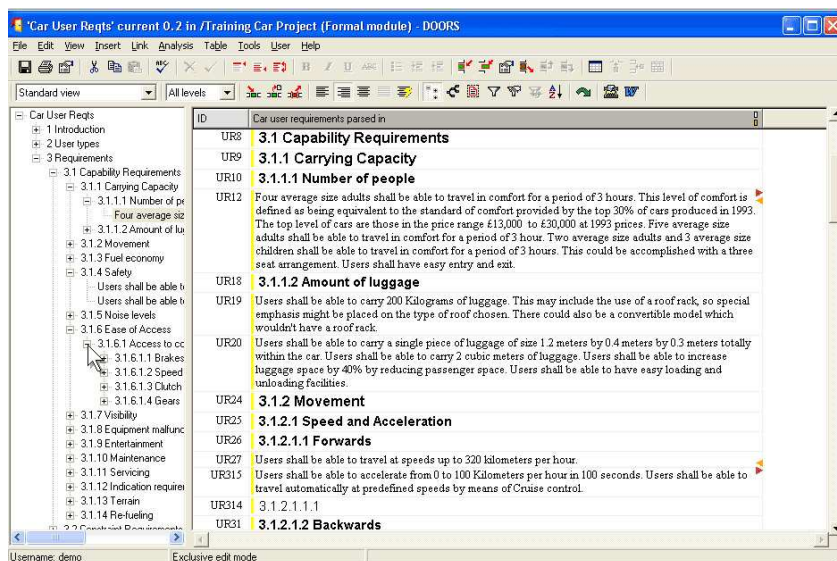


Abbildung 3: Screenshot Telelogic DOORS⁵

⁵ <http://www.telelogic.de>, Aufruf 08. 11. 2007

Simulink und Stateflow

Simulink und Stateflow sind auf Matlab basierende Erweiterungen von MathWorks.

Simulink dient zur zeitgesteuerten Simulation und Modellierung von Systemen, beispielsweise physikalischer, technischer oder (finanz-) mathematischer Art. Verwendet wird eine grafische Block-Darstellung mit Verbindungslinien für die Datenflussdarstellung. Ein Basissatz von Schaltblöcken wird mitgeliefert und weitere Blöcke sind von verschiedenen Herstellern käuflich erwerbbar. Stateflow kann integriert mit Simulink benutzt werden und wird zur ereignisgesteuerten Simulation eingesetzt. Mit Stateflow ist es möglich, komplexe Logik in Matlab zu beschreiben und hierarchische sowie parallele Zustände sowie die ereignisgesteuerten Übergänge zwischen diesen grafisch darzustellen. Hierdurch sind auch verschiedene Tests (u.a. zur Datenkonsistenz, der Wertebereiche, möglicher Endlosschleifen und auf fehlerhafte Zustands-Übergänge) zur Laufzeit möglich.

Targetlink

Targetlink ist ein Seriercodegenerierer für Simulink- und Stateflow-Modelle aus dem Hause dSPACE. Unterstützt werden Ansi-C, prozessoroptimierter Code und AUTOSAR. Mit Targetlink sind in der Simulationsumgebung Tests die drei Testmodi Model-In-The-Loop, Software-In-The-Loop und Processor-In-the-Loop möglich (siehe Vortrag über Testverfahren).

Targetlink benötigt eine laufende Matlab + Simulink Installation sowie das verwendete Blockset und läuft in der aktuellen Version 2.2 nur unter Microsoft Windows 2000 und XP.

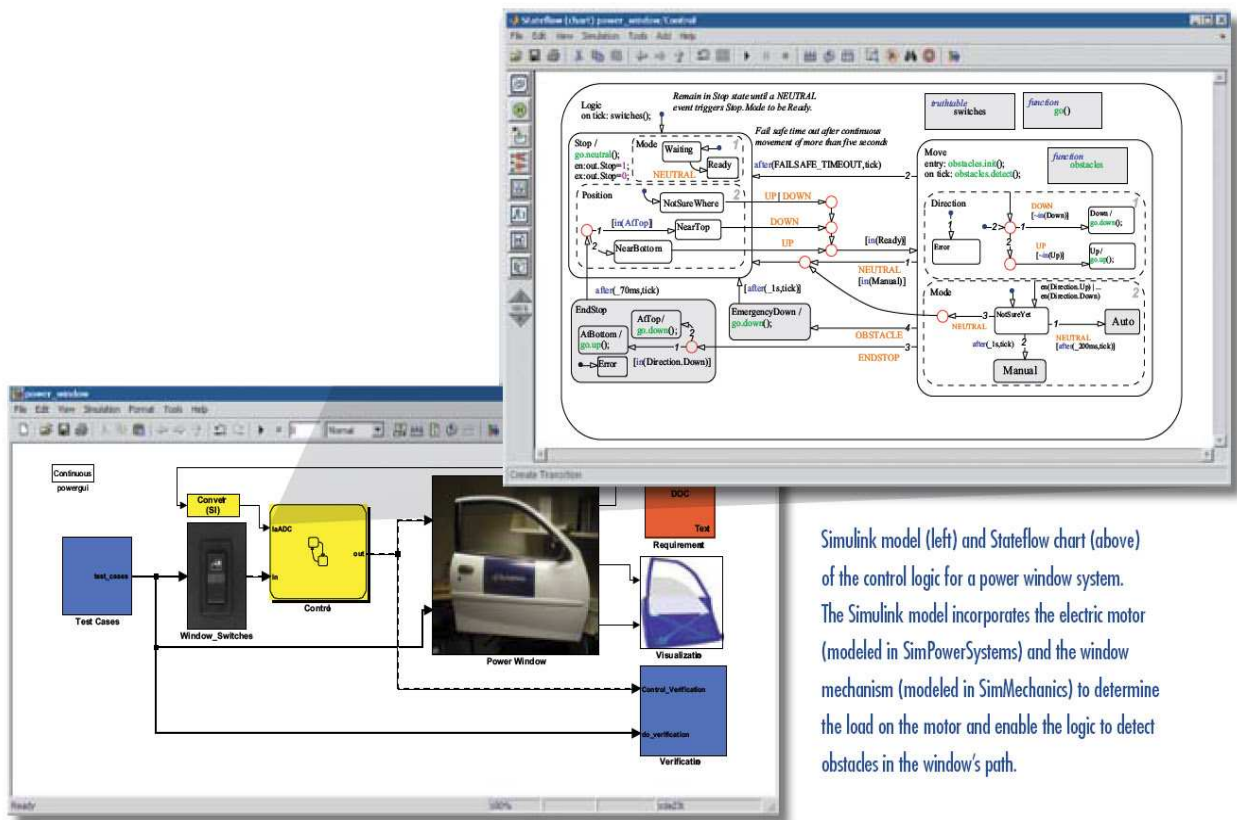


Abbildung 4: Screenshot Stateflow / Simulink⁶

3. Literatur

Jörg Schäuffele, Thomas Zurawka: Automotive Software Engineering, ATZ/MTZ-Fachbuch, Vieweg Verlag, Wiesbaden, 2006

Ottmar Beucher: MATLAB und Simulink, Pearson Studium, München, 2006

Sven Jangssen, Dr. Peter Krauss, Joachim Langenwalter, Armeé Shah (Hrsg.): International Automotive Conference 2004, Shaker Verlag, Aachen, 2004

⁶ Produktpräsentation MathWorks