



DER AUTOR

Alfred Vollmer ist
Redakteur der
elektronik industrie

(Nicht nur) die Zulieferer können aufatmen

Bisher mussten Zulieferer für jeden Automobil-Hersteller eine von Grund auf eigene Software-Version entwickeln. Im Rahmen eines ITEA-Projekts wird derzeit eine offene Software-Plattform (weiter-)entwickelt, die nach dem Middleware-Konzept arbeitet. Während die Basis-Software gleich bleibt erfolgt die Anpassung an die individuellen Bedürfnisse des Automobil-Herstellers erst in der oberen Software-Schicht – eine Vorgehensweise, die nicht nur bei der ganz besonders testintensiven Software für X-by-Wire-Funktionalitäten von großem Vorteil ist und langfristig die Rahmenbedingungen zur Entwicklung, Produktion und Wartung von Autos ganz entscheidend verändern kann.

Da die Verkaufspreise für Fahrzeuge auf Grund elektronischer Innovationen nicht stark steigen dürfen und andererseits die Kosten für die Software immer weiter in die Höhe schnellen, sah sich die Industrie gezwungen, ihre Kräfte zu bündeln, um nicht parallel jedes mal neu das Rad zu erfinden. Deshalb arbeiten sieben europäische Automobil-Hersteller und acht der führenden Zulieferer gemeinsam mit neun Forschungs-Unternehmen an dem ITEA-Projekt (Information Technology For European Advancement) Nr. 00009 mit der Bezeichnung EAST-EEA (Embedded Electronic Architecture), das ein Gesamtvolumen von 250 Mannjahren und ein Budget von 40 Millionen Euro aufweist.

Ziel des Projektes war es zunächst, die Kosten der Software-Entwicklung zu senken – und zwar durch Schaffung einer europaweit akzeptierten Embedded-Software-Architektur zur Informationsverarbeitung im Automotive-Bereich.

Die Herausforderung des zunächst vom 1.7.2001 bis zum 30.6.2004 laufenden

Projekts besteht darin, unterschiedliche elektronische Systeme, Subsysteme, Module und Komponenten zu integrieren, die von verschiedenen Lieferanten in ein Automotive-Gesamt-Netzwerk geliefert werden. Mit Hilfe von Hochsprachen sollen die Auto-Entwickler in der Lage sein, neue Funktionen zu implementieren sowie die praktische Seite einer neuen Gesetzgebung um zu setzen – und das alles durch Nutzung der bereits bestehenden Hard- und Firmware, selbst nach dem Verkauf sowie im Rahmen der Wartung.

Ziel von EAST-EEA ist es somit, eine angemessene Integrations-Plattform für die Automobil-Elektronik zu schaffen, was über die Definition einer offenen System-Architektur erfolgt, um so Hardware- und Software-Interoperabilität zu erzielen, so dass die meistverbreitete Hardware wieder verwendet werden kann.

Durch dieses grundlegende Konzept wird vermieden, dass jeder Automobil-Hersteller eigene Integrations-Frameworks (etwa: Rahmenbedingungen) entwickelt, wie sie zur Kommunikation, für Software-Schnittstellen sowie für Werkzeug-Umgebungen etc. benötigt werden.

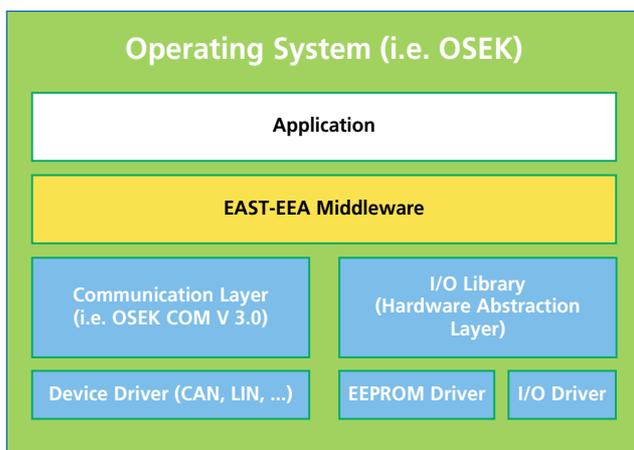
Für die Automobil-Hersteller, Zulieferer sowie für die Anbieter von Entwicklungs-Werkzeugen (inkl. Software) ergeben sich dadurch mehrere Vorteile wie die Verkürzung der Entwicklungszeit und die damit verbundene kürzere Time-to-Market, während sich gleichzeitig die Qualität

der Software erhöht. Darüber hinaus lassen sich Software-Komponenten erneut verwenden, während die Anzahl der Subsystem-Varianten geringer wird und Schnittstellen für Tools definiert werden.

Grundprinzip von EAST-EEA ist es, eine mehrschichtige Software-Architektur zu definieren, die auf das Middleware-Konzept (also ein Konzept mit einer Zwischenschicht) ausgerichtet ist. Diese Middleware bietet dann Schnittstellen und Dienste zur Unterstützung der Portabilität von qualitativ sehr hochwertigen Embedded-Software-Modulen an.

Dabei werden sowohl Standard-Aspekte als auch hochspezifische Aspekte proprietärer Anwendungen berücksichtigt. Gleichzeitig bezieht die offene Architektur bereits existierende Standards in den Bereichen Software-Entwicklung sowie Embedded-System-Architekturen mit ein.

Aus der Grafik geht die prinzipielle Positionierung bzw. die Einbettung der Middleware innerhalb eines Gesamtsystems hervor. Die Middleware-Schicht bietet der Anwendungsschicht (Application Layer) API-Dienste (API: Application Programming Interface; Schnittstelle zur Anwendungs-Programmierung), um so ein transparentes Mapping und Interaktionen zwischen unterschiedlichen Anwendungsfunktionen innerhalb des Fahrzeugs zu ermöglichen. Die Kommunikationsschicht (Communication Layer) offeriert der Middleware grundlegende Kommunikations-Dienste, die über Device-Driver (individuelle Treiber für einzelne Elemente) an die Fahrzeug-Netzwerke angepasst werden können.



Einbettung des Middleware-Konzepts in ein Gesamtsystem Grafik: EAST-EEA

Die Teilnehmer des Projekts EAST-EEA

Industrie

Audi (D), BMW (D), Centro Ricerche Fiat (I), DaimlerChrysler (D), ETAS (D), IXFIN Magneti Marelli Sistemi Elettronici (I), Fiat-GM Powertrain (D), PSA Peugeot Citroën Automobiles (F), G. I. E. Renault Recherche Innovation (F), Robert Bosch (D), Siemens SBS C-LAB (D), Siemens VDO Automotive (F, D), Valeo Electronique et Systèmes de Liaison (F), VECTOR Informatik (D), Volvo Technology (S), ZF Friedrichshafen (D)

Forschung

CEA-LIST (F), IRCCyN (F), INRIA (F), Linköping University of Technology (S), LORIA (F), Mälardalen University (S), Universität Paderborn – C-LAB (D), Royal Institute of Technology (S), Technische Universität Darmstadt (D)

Weitere fundamentale Bausteine innerhalb des Projekts EAST-EEA sind die System-Entwicklung und der Validierungs-Prozess von Architektur-Modulen sowie von funktionalen Modulen. Dabei bezieht EAST-EEA den gesamten Entwicklungsprozess auf Software- und System-Ebene mit ein, so dass der gesamte Prozess von der Anforderungs-Spezifikation über Entwicklung, Simulation und Implementations-Phasen in allen wesentlichen Aspekten abgedeckt wird, um so schließlich das funktionale und integrale Testen zu ermöglichen bzw. zumindest mit zu berücksichtigen.

Zum Erreichen dieser Ziele wird derzeit das sogenannte 4V-Modell in Kombination mit einer speziellen Architektur-Beschreibungssprache namens EAST-ADL (ADL: Architectur Description Language) entwickelt. Aufgabe der ADL ist es nun, zu garantieren, dass die

System- und Software-Spezifikationen auf einer angemessenen Abstraktions-Ebene gehalten werden. Außerdem unterstützen sowohl der Software-Entwicklungsprozess als auch die ADL nicht nur die Rückverfolgung (Trace-

ability) von Anforderungen zwischen frühen Anforderungs-Spezifikationen und Software-Komponenten, in denen die System-Funktionalität implementiert ist, sondern auch die Rückverfolgbarkeit zwischen zwei Phasen innerhalb des 4V-Modells.

Um sicher zu stellen, dass die funktionalen sowie die Architektur-Implementationen bestmöglich ihren spezifizierten Anforderungen entsprechen, werden sämtliche Phasen innerhalb des Entwicklungs-Prozesses sowie deren Ergebnisse jeweils einer V&V (Validierung und Verifikation) genannten Prozedur unterzogen.

Zur praktischen Validierung kommen Labor-Demonstratoren zum Einsatz, die den elektronischen Serienprodukten so nahe wie möglich kommen.

www.east-eea.net

300