

Webasto AG: UML-Software-Modellierung für die Fahrzeugmodelle von morgen.



Überblick

■ Die Aufgabe

Software-Entwicklung für fahrzeug- und komponentenspezifische elektronische Kontrolleinheiten in komplexen Bordnetzen unter Zeit- und Kostendruck bei höchsten Qualitäts- und Sicherheitsansprüchen

■ Die Lösung

Wechsel zur Software-Modellierung mit UML (Unified Modeling Language) und zu den Werkzeugen IBM Rational Rose Technical Developer und RealTime für die automatisierte Code-Generierung für eingebettete und Echtzeitsysteme

■ Die Vorteile

Reduzierung der Komplexität, Transparenz und Effizienz der Entwicklungsprozesse und der Kommunikation der beteiligten Spezialisten, schonender Ressourceneinsatz, hoher Investitionsschutz

Webasto: Komfort und Erlebnis in der Mobilität.

Wer früher Schiebedach oder Standheizung fürs Auto sagte, meinte Webasto. Wer heute Cabriodachsysteme, Global Comfort Solutions, Thermo-technik für Fahrzeuge aller Art, Brennstoffzellen für den automobilen und stationären Einsatz sagt, meint ebenfalls Webasto. Die Webasto AG in Stockdorf/München ist ein international operierender Zulieferer der Automobilindustrie und ihr 'Total Process Partner'. Das Unternehmen befindet sich seit der Gründung 1901 in Familienbesitz. Mit ca. 6 200 Mitarbeitern weltweit wurde 2005 ein Umsatz von ca. 1,4 Mrd. Euro erzielt, das sind 8,5 % mehr als im Vorjahr. Dabei wird das wirtschaftliche Umfeld immer anspruchsvoller: gestiegene Rohstoffkosten, harter Wettbewerb, Preisdruck.

Tempo vorgeben statt mitgehen.

Um permanent innovative Produkte zeitnah zur Serienreife zu bringen, investiert Webasto jedes Jahr hohe zweistellige Millionenbeträge in die

Forschung und Entwicklung. Neben den richtigen strategischen Weichenstellungen ist es entscheidend für den Erfolg, alle Einsparpotenziale auf Produkt- und auf Prozessebene zu nutzen. „Wir wollen beim Tempo nicht nur mitgehen, sondern es vorgeben“, sagt Stefan Schober, Team Manager Software Development bei Webasto. „Die Entwicklungszeiten werden immer kürzer. Gleichzeitig verlangen die Kunden, dass die Zeiten ohne Kostensteigerungen oder Abstriche bei der Qualität eingehalten werden.“

Erfolgsfaktor Software-Entwicklung.

Beispiel Cabriodachsysteme: Jedes Dachsystem ist eine spezifische Entwicklung, die hohe Vorleistungen erfordert. Innerhalb des Netzwerks des Fahrzeugs kommuniziert das Steuerungssystem des Verdecks mit vielen anderen Komponenten wie Scheibenheber, Heckschloss, Kofferraum oder Türen, um den gesamten Ablauf des Öffnens oder Schließens zu steuern.

Die Software-Entwicklung der sogenannten 'Electronic Control Units' (kurz ECUs) ist deshalb ein kritischer Erfolgsfaktor. Zwischen 60 und 70 ECUs kontrollieren von der Klimaanlage über die elektrischen Fensterheber bis hin zum automatischen Türöffner alles, was ein modernes Fahrzeug heute an Komfort zu bieten hat. Mit steigender Geräteanzahl wird das Fahrzeugnetz immer komplexer.

Kommunikation auf allen Ebenen.

Nicht nur die unterschiedlichen ECUs dieser Komponenten müssen reibungslos miteinander kommunizieren, sondern auch die Hersteller der unterschiedlichen Bauteile und andere Beteiligte des Fertigungsprozesses.

„Die Steuerungs-Software muss auf alle Ansprüche der vernetzten Technologien reagieren können. Deshalb ist es notwendig, dass alle Spezialisten diese möglichst genau verständlich machen können. Es muss transparent sein, was die Software bewirkt“, so Stefan Schober. „Wir benötigten daher eine Kommunikationsplattform, die für alle Beteiligten – also auch für diejenigen ohne explizites Entwickler-Know-how – die relevanten Abläufe des Systems sichtbar macht. Die Lösung sollte der Komplexität der Aufgabe gewachsen sein, sich aber nicht zu sehr in Details verlieren dürfen.“

Werkzeug für Embedded- und Echtzeit-Betrieb gesucht.

Für Webasto erwies sich die Modellierung mit UML (Unified Modeling Language) als adäquate Darstellungsart für ihre Projekte. Stefan Schober prüfte zusammen mit seinem Team mehrere UML-Werkzeuge. „Maßgeblich für unsere Entscheidung war dabei, dass das Werkzeug für den Echtzeit- und Embedded-Bereich geeignet war.“ Als beste Lösung erwies sich das Werkzeug IBM Rational Rose Technical Developer. Es war auf Anforderungen eingebetteter Systeme spezialisiert und unterstützte bereits das Konzept von Capsules, einem Design-Pattern für aktive Klassen, das im Rahmen der UML 2.0 jetzt unter dem Namen 'Structured Classes' bekannt geworden ist. Außerdem überzeugte das Preis-Leistungs-Verhältnis und die Sicherheit eines führenden Herstellers in Bezug auf Software-Pflege und -Weiterentwicklung. Das bedeutet Investitionsschutz und Zukunftssicherheit.

Modellierung mit UML und automatische Code-Generierung.

Die UML-Modellierung reduziert die Komplexität und beschleunigt die Entwicklung. Darüber hinaus steht sie für den Paradigmenwechsel vom prozeduralen Konzept hin zu einem objektorientierten Ansatz und ermöglicht Code-Generierung für die nicht objektorientierte Programmiersprache C. Die Modellierung mit UML ist eine wichtige Disziplin in der Software-Entwicklung, ebenso die zunehmende automatisierte Code-Generierung im Rahmen von Model Driven Development (MDD), die erfahrungsgemäß die Effizienz in der Entwicklung steigert und durch Vermeidung von Fehlern die Qualität verbessert. Stefan Schober: „Für einzelne Hardwarekomponenten ist die Programmierung 'von Hand' mit der Programmiersprache C ausreichend, nicht aber für die funktionsbestimmende Software. Diesen Teil, der etwas mehr als 50 Prozent unserer Arbeit in Anspruch nimmt, modellieren wir heute mit Rational Rose RealTime.“

Funktionalität, Wirtschaftlichkeit und Bedienkomfort überzeugen.

Durch die planvolle, bildliche Darstellung am PC bis hin zur Simulation und der Voraussicht, wie die beteiligten Elemente interagieren, ermöglicht Rational eine viel effizientere Kommunikation zwischen den Mitarbeitern. Die automatische Transformation des Modells in einen ablauffähigen Code senkt den Zeitaufwand für die Entwickler bei Webasto. Zudem kann das Unternehmen wertvolle Speicherressourcen schonen, sodass hier trotz steigender Ansprüche keine zusätzlichen Investitionen erforderlich sind. Letztlich zeichnet sich das Werkzeug auch durch leichte Bedienbarkeit und ein breites Kompatibilitätsspektrum aus. „Wir setzen IBM Rational Rose seit Ende 2002 ein und sind sehr zufrieden“, lautet das Fazit von Stefan Schober.

„Wir sparen viel Zeit und Aufwand durch die UML-Modellierung. Auch Spezialisten ohne explizites Entwickler-Know-how werden optimal eingebunden.“

Stefan Schober, Team Manager Software Development, Webasto AG, Stockdorf



IBM Deutschland GmbH
70548 Stuttgart
ibm.com/de

IBM Österreich
Obere Donaustraße 95
1020 Wien
ibm.com/at

IBM Schweiz
Vulkanstrasse 106
8010 Zürich
ibm.com/ch

Die IBM Homepage finden Sie unter:
ibm.com

IBM, das IBM Logo und ibm.com sind eingetragene Marken der IBM Corporation.

Rational ist eine Marke der IBM Corporation in den USA und/oder anderen Ländern.

Weitere Unternehmens-, Produkt- oder Servicennamen können Marken anderer Hersteller sein.

Diese Erfolgsgeschichte verdeutlicht, wie ein bestimmter IBM Kunde Technologien/Services von IBM und/oder einem IBM Business Partner einsetzt. Die hier beschriebenen Resultate und Vorteile wurden von zahlreichen Faktoren beeinflusst. IBM übernimmt keine Gewährleistung dafür, dass in anderen Kundensituationen ein vergleichbares Ergebnis erreicht werden kann. Alle hierin enthaltenen Informationen wurden vom jeweiligen Kunden und/oder IBM Business Partner bereitgestellt. IBM übernimmt keine Gewähr für die Richtigkeit dieser Informationen.

Gedruckt in Deutschland.

© Copyright IBM Corporation 2006
Alle Rechte vorbehalten.