

Neue Wege zur Steuergeräte-Software

AUTOSAR in der Praxis – Der Lebenszyklus von AUTOSAR-Software (Teil 2)

In Teil 1 dieses Artikels [1] wurden der Aufbau einer AUTOSAR-konformen Steuergeräte-Software sowie die AUTOSAR-Entwicklungsmethode dargestellt. Teil 2 zeigt nun, wie eine AUTOSAR-Steuergeräte-Software während ihres Lebenszyklus gepflegt wird. Um die typischerweise in einem Entwicklungsprojekt auftretenden Änderungen handhaben zu können, ist eine gute Werkzeugunterstützung essentiell.

Von Von Pascale Morizur, Matthias Wernicke und Justus Maier

Während der Entwicklung eines AUTOSAR-Steuergerätes werden Basis-Software (BSW) und Runtime Environment (RTE) durch Code-Generatoren an die spezifischen Anforderungen des Steuergerätes angepasst. Grundlage für das Generieren sind die umfangreichen Beschreibungsdateien, die bereits in Teil 1 des Artikels erwähnt wurden:

- ▶ Der „ECU Extract of System Description“

enthält den steuengerätespezifischen Ausschnitt des Gesamtsystems.

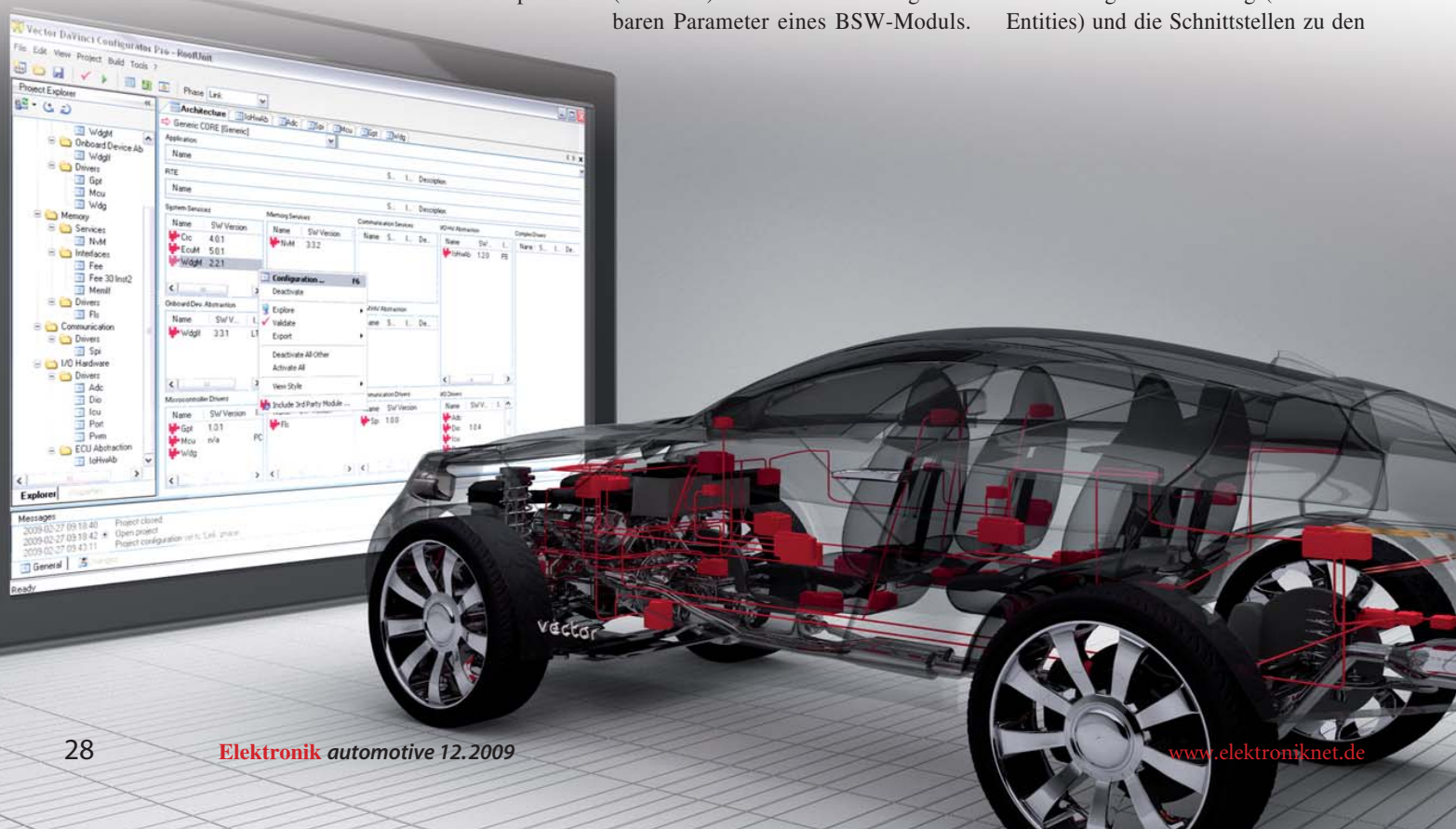
- ▶ Die „SWC Description“ beschreibt die Schnittstellen und Struktur der AUTOSAR-Software-Komponenten und kann bereits im „ECU Extract of System Description“ enthalten sein.
- ▶ Die „ECU Configuration Description“ beinhaltet die Konfiguration der BSW-Module und der RTE.

- ▶ Die „BSW Module Description“ (BSWMD) beschreibt die konfigurierbaren Parameter eines BSW-Moduls.

Die Herausforderung für den Entwickler besteht darin, diese Beschreibungsdateien konsistent zu halten, ohne bei jeder Änderung die gesamte Konfiguration neu erstellen zu müssen. Bei dieser Aufgabe helfen Konfigurationswerkzeuge, die iterative Arbeitsabläufe während eines Steuergeräte-Entwicklungsprojektes unterstützen. Typische Szenarien und Änderungsfälle werden im Folgenden beschrieben und die Arbeitsschritte anhand der Vector-Werkzeuge DaVinci Developer und DaVinci Configurator Pro erklärt.

■ Neuentwicklung von Software-Komponenten

Je nach OEM kann der „ECU Extract of System Description“ als Vorgabe bereits die Schnittstellen von Software-Komponenten enthalten. In manchen Fällen sind diese SWCs noch nicht vollständig. Es werden z.B. die Applikationsschnittstellen vom OEM vorgegeben, nicht jedoch die Implementierungsbeschreibung (Runnable Entities) und die Schnittstellen zu den



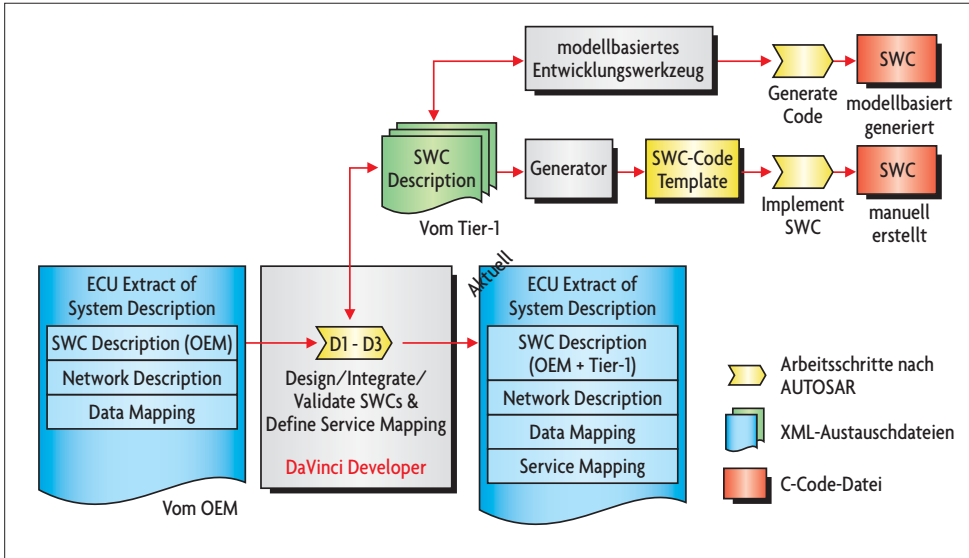


Bild 1. Entwicklung einer Funktions-Software, die aus mehreren SWCs besteht.

Standard-Services der BSW-Module. Diese fehlende Spezifikation ergänzt der Entwickler beim Tier-1-Zulieferer mit Hilfe eines Werkzeugs wie dem DaVinci Developer. Ebenso kann der Tier-1-Zulieferer eigene SWCs entwerfen (Schritt D1 in Bild 1).

Das Implementieren der SWCs kann auf zwei Arten erfolgen: Durch manuelles Implementieren auf Basis eines generierten Code-Templates oder mithilfe von modellbasierten Entwicklungswerkzeugen, wie z.B. Matlab/Simulink EmbeddedCoder oder Target-Link. Die SWC Description wird hierbei als Basis für das Modellieren der SWC importiert. Durch Code-Generatoren wird die Implementierung der SWCs automatisch erstellt.

Falls beim Tier-1-Zulieferer bereits SWCs existieren, können diese ebenfalls ins Steuergerät integriert werden. Hierzu wird die zugehörige SWC Description importiert und die SWC mit den anderen SWCs des Steuergerätes verbunden (Schritt D2).

In einem weiteren Integrations-schritt wird zudem die Zuordnung der Datenelemente der SWC-Schnittstellen zu den Bussignalen definiert (Data Mapping) – falls diese Zuordnung nicht schon im „ECU Extract of System Description“ durch den OEM definiert worden ist.

Typischerweise ändert sich der „ECU Extract of System Description“ mehrmals im Laufe eines Projektes. Wenn der Tier-1-Zulieferer eine neue Version erhält, können auch die darin

vorgegebenen SWCs geändert worden sein. Mit einer speziellen Aktualisierungsfunktion erlaubt der DaVinci Developer die Übernahme der Änderungen, ohne dass die durch den Tier-1-Zulieferer bisher vorgenommenen Erweiterungen wie Implementierungsbeschreibung oder Schnittstellen zu den Standard-Services verlohren gehen.

Konfiguration der Standard-AUTOSAR-Services

Eine Herausforderung beim Konfigurieren von AUTOSAR-Steuergeräten ist das Konfigurieren der Standard-Services der BSW-Module. Innerhalb des „ECU Extract of System Description“ sind die Services durch spezielle SWCs – die Service Components – repräsentiert. Deren Erstellung und die Integration mit den SWCs durch das Service Mapping kann werkzeugge-

stützt automatisch erfolgen (Schritt D3).

Im Bottom-up-Verfahren wird zunächst der Dienst im BSW-Modul konfiguriert, z.B. basierend auf Vorgaben des OEMs. Die SWCs werden anschließend passend zur Konfiguration des Services erweitert. Beispiel hierfür ist der DCM (Diagnostic Communication Manager).

Ähnlich wie die Services werden auch die I/O-Schnittstellen des Steuergerätes durch eine spezielle SWC repräsentiert – die „I/O Hardware Abstraction Component“. Auch diese SWC kann in einem Top-down- oder Bottom-up-Verfahren entstehen.

Am Ende der Integration hat der Tier-1-Zulieferer einen aktualisierten „ECU Extract of System Description“. Zusätzlich zur ursprünglichen Version des OEMs enthält die Datei nun zusätzlich die vom Tier-1-Zulieferer definierten Teile und ist vollständig validiert.

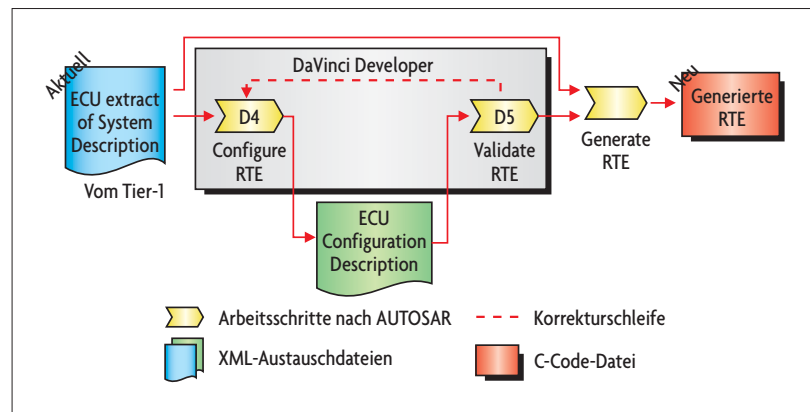


Bild 2. Arbeitsschritte für die Konfiguration der RTE.

■ Anpassung der RTE nach dem Ändern von SWCs

Falls sich lediglich die Implementierung einer SWC ändert, ohne dass die Schnittstellen oder die Struktur der SWC betroffen sind, muss weder die RTE noch die Basis-Software neu konfiguriert werden. Falls sich allerdings die Struktur einer SWC ändert, z.B. durch Hinzufügen eines neuen Runnable Entity, muss die RTE-Konfiguration angepasst werden. Dabei wird

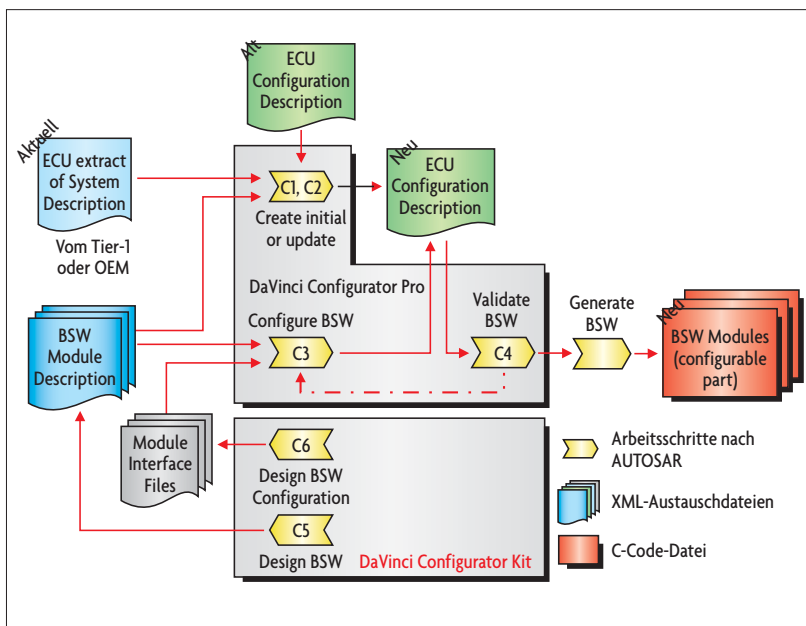
kommunikationsbezogenen BSW-Module relevant und erfordert keine Änderung der RTE oder der SWCs.

Bild 3 zeigt die Arbeitsschritte, die zur Übernahme der geänderten Kommunikationsdaten durchgeführt werden. Dieses Anpassen der BSW-Module kann automatisch erfolgen: Zunächst wird eine neue „ECU Configuration Description“ erzeugt und der Inhalt der alten in die neue „ECU Configuration Description“ transformiert (Schritte C1, C2). Alle Parame-

■ Austausch des Prozessorderivats oder der Prozessorfamilie

Dank der konsequenten Hardware-Abstraktion durch AUTOSAR muss der Tier-1-Zulieferer beim Wechsel auf ein anderes Prozessorderivat oder eine andere Prozessorfamilie lediglich die betroffenen MCAL-Module austauschen. Dieser Austausch wird werkzeuggestützt durchgeführt, indem die alten MCAL-Module entfernt und die neuen MCAL-Module durch Auswahl der BSWMD-Dateien zum Projekt hinzugefügt werden. Die Werte der neu hinzugekommenen, plattformabhängigen Parameter werden im DaVinci Configurator Pro überprüft und vom Tier-1-Zulieferer gegebenenfalls nachkonfiguriert (Schritt C3 für jedes geänderte MCAL-Modul). Durch das finale Validieren (Schritt C4) ist die Konsistenz der „ECU Configuration Description“ wieder hergestellt.

Besonders nützlich für den Tier-1-Zulieferer ist die Erweiterbarkeit der Werkzeugumgebung, um z.B. MCAL-Module zukünftiger Prozessoren oder eigene BSW-Module unterstützen zu können. Dies ermöglicht das DaVinci Configurator Kit, mit dem der Tier-1-Zulieferer BSWMD-Dateien und „Module Interface“-Dateien erstellen kann. Diese enthalten die Definitionen für spezifische Konfigurationsoberflächen, Validierungsregeln oder Generatoren für die BSW-Module (Schritte C5 und C6).



I Bild 3. Arbeitsschritte für die Konfiguration der Basis-Software.

das neu hinzugekommene Runnable Entity einer Task zugeordnet. Nach dem Anpassen (Schritt D4 in Bild 2) wird die RTE-Konfiguration validiert (Schritt D5).

Der DaVinci Developer unterstützt diesen Prozess durch detaillierte Fehlermeldungen und Korrekturvorschläge. Es werden z.B. Inkonsistenzen gemeldet, damit der Tier-1-Zulieferer die SWC Description oder die RTE-Konfiguration korrigieren kann.

■ Geänderte Kommunikationsdaten des OEMs übernehmen

Ein typisches Änderungsszenario ist die Aktualisierung der Kommunikationsdaten eines Steuergeräts, z.B. aufgrund einer geänderten Verteilung der Signale auf die Botschaften. Eine derartige Änderung ist nur für die kom-

terwerte, die nicht von der Änderung betroffen sind, werden dadurch automatisch übernommen. Lediglich die Parameter der neuen oder geänderten Signale oder Botschaften werden anschließend im Schritt C3 konfiguriert. Um sicherzustellen, dass alle Parameterwerte konsistent sind, wird die aktuelle „ECU Configuration Description“ schließlich im Schritt C4 validiert.

Vector hat sowohl die Aktualisierungsfunktion als auch das Validieren als Ergänzung des AUTOSAR-Standards im DaVinci Configurator Pro implementiert. Sollte der OEM keinen „ECU Extract of System Description“ gemäß AUTOSAR liefern, kann der Tier-1-Zulieferer stattdessen die gewohnten Netzwerk-Beschreibungsformate (DBC, LDF oder FIBEX) als Eingangsformat für die DaVinci-Werkzeuge verwenden.

■ Austausch der OEM-spezifischen Diagnose

Soll eine bestehende Steuergeräte-Konfiguration für einen anderen OEM übernommen werden, muss die Diagnose-Basis-Software des Steuergerätes ausgetauscht werden, weil diese OEM-spezifisch ist. Dies betrifft die BSW-Module DCM und DEM (Diagnostic Event Manager).

Bild 4 zeigt, wie dieser Austausch erfolgt. Der Tier-1-Zulieferer erhält vom OEM die neue CDD- oder ODX-Datei. Diese weit verbreiteten Formate enthalten die Diagnosebeschreibungsdaten des Steuergeräts und können mit Werkzeugen wie CANdela Studio von Vector erzeugt werden.

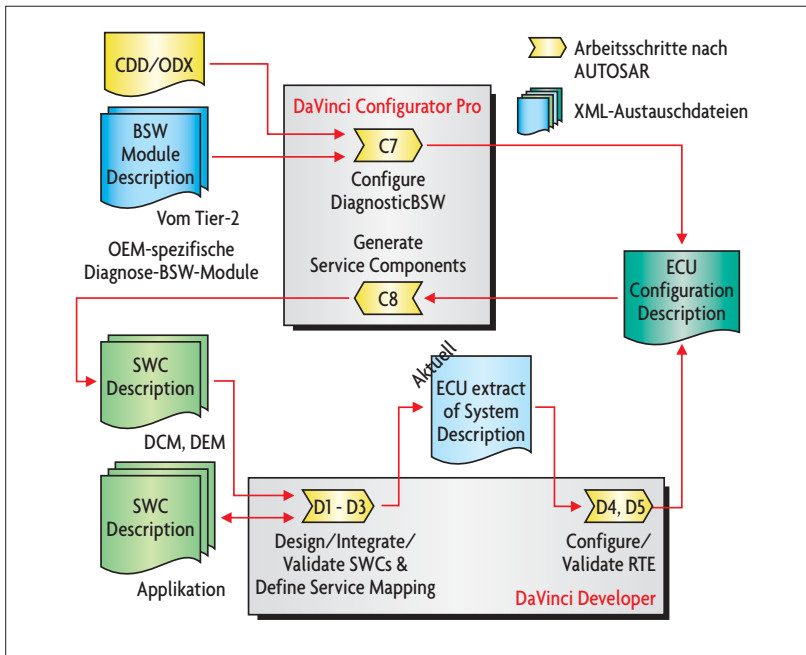


Bild 4. Die Arbeitsschritte beim Anpassen der diagnose-spezifischen Software-Teile eines Steuergeräts.

Die Informationen aus diesen Dateien verwendet der DaVinci Configurator Pro zum automatischen Konfigurieren der Diagnose-BSW-Module des Steuergeräts (Schritt C7). Analog zu den geänderten Diagnose-BSW-Modulen müssen auch die DCM und DEM Service Components angepasst und in einem Bottom-up-Prozess den Applikations-SWCs bekannt gemacht werden. Dazu generiert der DaVinci Configurator Pro aus der „ECU Con-

figuration Description“ „SWC Description“-Dateien mit den Service-Komponenten für DCM und DEM (Schritt C8).

Mit Hilfe des DaVinci Developer kann der Tier-1-Zulieferer nun das Service Mapping für alle Diagnose-Services des neuen OEMs erzeugen. Durch die Validierung der SWCs ist sichergestellt, dass dabei kein Dienst vergessen wird. Falls die Applikations-SWCs die vom OEM geforder-

ten Diagnose-Services noch nicht anbieten, müssen sie erweitert werden (Schritte D1 bis D3), was wiederum ein Anpassen der RTE erfordert (Schritte D4, D5).

■ Ändern von I/O-Signalen

Soll an das Steuergerät zum Beispiel ein neuer Sensor angeschlossen werden, muss das dafür verwendete neue I/O-Signal in die „ECU Configuration Description“ eingefügt werden. Hierzu erweitert der Tier-1-Zulieferer im DaVinci Configurator Pro die Konfiguration der „I/O Hardware Abstraction“ um das neue Signal (Schritt C3 in Bild 3) und passt z.B. das Pin-Mapping in den I/O-Treibern in der MCAL-Schicht an. Nach dem erfolgreichen Validieren dieser Konfigurationsänderung wird eine aktualisierte SWC zur Repräsentation der „I/O Hardware Abstraction“ generiert. Durch den Import dieser SWC in den DaVinci Developer kann der Tier-1-Zulieferer die SWCs der Funktions-Software für den Zugriff auf den neuen Sensor erweitern. sj

Literatur

- [1] Morizur, P.; Wernicke, M.; Maier, J.: Neue Wege zur Steuergeräte-Software – Teil 1. *Elektronik automotive* 2009, H. 11, S. 38ff.



**Dipl.-Ing.
Pascale Morizur**

studierte Physik-Elektronik an der Grande Ecole ICPI in Lyon (F). Nach ihrem Abschluss 1986 arbeitete sie zehn Jahre bei MAN-Nutzfahrzeuge in der Vorentwicklung im Bereich CAN, J1939 und Diagnose. Seit 2005 ist sie bei Vector als Produkt-Managerin im Bereich Embedded-Software-Komponenten tätig.

pascale.morizur@vector.com



**Dipl.-Ing. (FH)
Matthias Wernicke**

war nach Abschluss seines Studiums der Industrieelektronik an der FH Ulm zunächst vier Jahre im Daimler Forschungszentrum in Ulm tätig. Seit Anfang 2000 arbeitet er bei Vector Informatik in Stuttgart an der Entwicklung von Methoden und Werkzeugen für den Entwurf verteilter Elektronikfunktionen im Kfz. Er ist heute für das Produkt-Management der DaVinci-AUTOSAR-Werkzeuge verantwortlich.

matthias.wernicke@vector.com



**Dipl.-Inf. (FH)
Justus Maier**

studierte Informatik in Regensburg. Er begann seine berufliche Laufbahn als Entwickler für Standard-Software in der Versicherungsbranche. Seit acht Jahren beschäftigt er sich mit der Konzeption und Weiterentwicklung von Werkzeugen für die Steuergerätekonfiguration im AUTOSAR-Umfeld. Als technischer Produkt-Manager des DaVinci Configurator Pro ist er bei Vector seit 2006 tätig.

justus.maier@vector.com