



## BMBF Verbundprojekt *D-MINT*

# **Entwicklung und Erprobung innovativer Technologien des modellbasierten Tests**

### **Projektpartner**

**Fraunhofer FOKUS, Nokia Siemens Networks, PikeTec,  
Daimler AG, Fraunhofer IESE, Testing Technologies,  
ABB AG, iXtronics, INSPIRE AG**

### **Schlussbericht für die Projektlaufzeit vom 1.4.2007 bis zum 31.12.2009**

*Das diesem Bericht zugrunde liegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter den Förderkennzeichen 01IS07001A, 01IS07001B, 01IS07001C, 01IS07001D, 01IS07001E, 01IS07001F, 01IS07001G und 01IS07001H gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt bei den Autoren.*

**Ina Schieferdecker, Axel Rennoch, FhG FOKUS;  
Thomas Deiß, Colin Willcock, Nokia Siemens Network;  
Eckard Bringmann, Piketec; Jens Herrmann, Daimler; Thomas Bauer, FhG IESE;  
Stephan Pietsch, Testing Technologies; Detlef Streitferdt, ABB  
Ulrich Meier-Noe, iXtronics; Andreas Weitkamp, INSPIRE**

## **Inhaltsverzeichnis**

<b>1. Aufgabenstellung .....</b>	<b>5</b>
<b>2. Voraussetzungen, unter denen das Vorhaben durchgeführt wurde .....</b>	<b>6</b>
<b>3. Planung und Ablauf des Vorhabens.....</b>	<b>8</b>
<b>4. Wissenschaftlicher und technischer Ausgangsstand .....</b>	<b>9</b>
<b>5. Zusammenarbeit mit anderen Stellen.....</b>	<b>11</b>
<b>6. Fraunhofer FOKUS, Förderkennzeichen 01 IS 07 001 A .....</b>	<b>12</b>
6.1 Erreichte Ergebnisse.....	12
6.1.1 Thematische Ergebnisse .....	12
6.1.2 Organisatorische Arbeiten .....	17
6.2 Voraussichtlicher Nutzen .....	18
6.3 Fortschritt bei anderen Stellen .....	18
6.4 Erfolgte oder geplante Veröffentlichungen .....	18
<b>7. Nokia Siemens Networks, Förderkennzeichen 01 IS 07 001 B.....</b>	<b>21</b>
7.1 Erreichte Ergebnisse.....	21
7.1.1 Thematische Ergebnisse .....	21
7.1.2 D-MINT Projektleitung .....	24
7.2 Voraussichtlicher Nutzen .....	24
7.3 Fortschritt bei anderen Stellen .....	24
7.4 Erfolgte oder geplante Veröffentlichungen .....	25
<b>8. Piketec, Förderkennzeichen 01 IS 07 001 C .....</b>	<b>26</b>
8.1 Erreichte Ergebnisse .....	26
8.1.1 Durchgängigkeit von Tests von MiL bis HiL.....	26
8.1.2 Unterstützung von Applikationsparametern in Testmodellen .....	27
8.1.3 Integration TPT mit CameLView.....	28
8.1.4 Fallstudie Daimler .....	29
8.1.5 Fallstudie ABB .....	30
8.2 Voraussichtlicher Nutzen .....	30
8.3 Fortschritt bei anderen Stellen .....	31
8.4 Erfolgte oder geplante Veröffentlichungen .....	31
<b>9. Daimler AG, Förderkennzeichen 01 IS 07 001 D .....</b>	<b>32</b>
9.1 Erreichte Ergebnisse .....	32
9.1.1 Inhaltliche Ergebnisse.....	32
9.1.2 D-MINT Arbeitspaketeleitung.....	35
9.2 Voraussichtlicher Nutzen .....	37
9.3 Fortschritt bei anderen Stellen .....	37
9.4 Erfolgte oder geplante Veröffentlichungen .....	38
<b>10. Fraunhofer IESE, Förderkennzeichen 01 IS 07 001 E.....</b>	<b>39</b>
10.1 Erreichte Ergebnisse.....	39
10.1.1 Thematische Ergebnisse .....	39

10.1.2	Organisatorische Arbeiten .....	45
10.2	Voraussichtlicher Nutzen .....	45
10.3	Fortschritt bei anderen Stellen .....	45
10.4	Erfolgte oder geplante Veröffentlichungen.....	45
<b>11.</b>	<b>Testing Technologies, Förderkennzeichen 01 IS 07 001 F .....</b>	<b>47</b>
11.1	Erreichte Ergebnisse.....	47
11.1.1	Modellbasiertes Testen mit UTP .....	47
11.1.2	UTP Umsetzung und Erweiterung.....	48
11.1.3	Fallstudie IDEKO/Solaruce.....	48
11.1.4	Fallstudie ABB .....	49
11.1.5	Integration TTmodeler/TTworkbench mit CAMEL-View.....	50
11.2	Fortschritt bei anderen Stellen .....	51
11.3	Erfolgte oder geplante Veröffentlichungen.....	51
11.4	Erfolgte oder geplante Veröffentlichungen.....	51
<b>12.</b>	<b>ABB AG, Förderkennzeichen 01 IS 07 001 G .....</b>	<b>52</b>
12.1	Erreichte Ergebnisse.....	52
12.1.1	Fallstudie .....	52
12.1.2	Anangepasste Modellbasierte Testmethode .....	54
12.1.3	Auswertung der Fallstudie .....	55
12.2	Voraussichtlicher Nutzen .....	56
12.3	Fortschritt bei anderen Stellen .....	56
12.4	Erfolgte oder geplante Veröffentlichungen.....	56
<b>13.</b>	<b>iXtronics GmbH, Förderkennzeichen 01 IS 07 001 H .....</b>	<b>57</b>
13.1	Erreichte Ergebnisse.....	57
13.1.1	Werkzeugkopplung mit PikeTec TPT .....	57
13.1.2	Werkzeugkopplung mit TestingTech TTWorkbench.....	58
13.1.3	Fallstudie ABB Softstarter .....	59
13.2	Voraussichtlicher Nutzen .....	60
13.3	Fortschritt bei anderen Stellen .....	60
13.4	Erfolgte oder geplante Veröffentlichungen.....	60
<b>14.</b>	<b>INSPIRE AG, Förderkennzeichen 01 IS 07 001 I .....</b>	<b>61</b>
14.1	Erreichte Ergebnisse.....	61
14.1.1	Verwertungsplanung .....	61
14.1.2	Produktpositionierung.....	65
14.1.3	Industrielle Metriken .....	69
14.2	Voraussichtlicher Nutzen .....	73
14.3	Fortschritt bei anderen Stellen .....	73
14.4	Erfolgte oder geplante Veröffentlichungen.....	73
<b>15.</b>	<b>Liste der Veröffentlichungen.....</b>	<b>74</b>
<b>16.</b>	<b>Veröffentlichungen Anderer Stellen/Ausserhalb des Projekts .....</b>	<b>79</b>
<b>17.</b>	<b>Abkürzungsverzeichnis .....</b>	<b>80</b>

 Deployment of Model-Based Technologies to Industrial Testing	<b>D-MINT Abschlussbericht</b>	Seite : 4 of 81 Version: 1.0 Datum: 30.06.2010 <hr/> Status : Final
---	--------------------------------	--

## Zusammenfassung

Im Rahmen des europäischen ITEA2-Forschungsprojekts D-MINT wurden innovative modellbasierte Testmethoden identifiziert, weiterentwickelt und in industriellen Fallstudien eingesetzt. Unterschiedliche Technologien wurden durch neue Testansätze ergänzt, aufeinander abgestimmt und in einem Framework zusammengefasst.

Die deutschen Projektpartner haben insbesondere durch folgende Arbeiten beigetragen:

- Erarbeitung eines Architektur-basierten Testableitungsverfahrens,
- Erstellung des model-basierten statistischen Testansatzes,
- der Nutzbarmachung des UML Testprofils,
- der Integration von ausgewählten Testwerkzeugen,
- mit einer eigenen Fallstudie im Bereich industrielle Automatisierung,
- mit zwei eigenen Fallstudien im Bereich Automotive,
- mit Beiträgen zu Fallstudien im Bereich der Telekommunikation,
- mit neuen Ansätzen und Werkzeugen zur Bewertung von Testqualitäten,
- mit der Entwicklung eines Prozesses zur Bewertung von MBT Verfahren.
- mit einer Marktanalyse zum Einsatz von MBT

Die deutschen Projektpartner sind Daimler AG (Stuttgart), Nokia Siemens Network (Düsseldorf), PikeTec (Berlin), ABB (Mannheim), Testing Technologies (Berlin), iXtronics (Paderborn), INSPIRE (Paderborn), Fraunhofer IESE (Kaiserslautern) und Fraunhofer FOKUS (Berlin).

Das europäische D-MINT Konsortium wurde durch den deutschen Projektpartner Nokia Siemens Network und das deutsche Konsortium von Fraunhofer FOKUS geleitet.

 Deployment of Model-Based Technologies to Industrial Testing	<b>D-MINT Abschlussbericht</b>	Seite : 5 of 81 Version: 1.0 Datum: 30.06.2010 Status : Final
---	--------------------------------	--

## Teil I: Gesamtprojekt

### 1. AUFGABENSTELLUNG

Das Projekt D-MINT wird innovative modellbasierte Testtechnologien in der Industrie zur Anwendung bringen und diese weiterentwickeln. Diese Technologien werden es ermöglichen, qualitativ hochwertige softwareintensive Systeme und Produkte in kürzerer Zeit und mit reduzierten Kosten auf den Markt zu bringen.

Mit dem Vordringen von softwareintensiven Systemen in immer mehr Bereiche unserer Gesellschaft wachsen die Anforderungen an die Funktionalität dieser Systeme. Mit wachsenden Anforderungen wächst auch die Komplexität dieser Systeme und damit wiederum der Zeitdruck für die Entwicklung und die Fehleranfälligkeit der Systeme. Um wettbewerbsfähig zu bleiben, sind leistungsfähige und kostengünstige Technologien für die Entwicklung und den Test einzusetzen. Modellbasierte Testtechnologien versprechen mehr Leistung und mehr Möglichkeiten zur Automatisierung und damit zur Kostenreduzierung, da es viele Modelltypen erlauben, Testfälle direkt und z.T. auch automatisch aus den Modellen abzuleiten.

In der Industrie werden einzelne Modelltypen bereits isoliert für die Entwicklung eingesetzt. In einzelnen Fällen wird auch bereits auf der Basis von Modellen getestet. Noch ist dieses Vorgehen jedoch auf einzelne Fälle beschränkt, oftmals wird der modellbasierte Test vor allem für den Test einzelner Einheiten (z.B. Komponenten) durchgeführt. Für den darauf aufsetzenden modellbasierten Integrations- und den sich daran anschließenden Systemtest, die beide in der Lage sein müssen, auf verschiedenen Modelltypen (inkl. der Kombination verschiedener Modelle) aufzusetzen, besteht in der Industrie noch großer Handlungsbedarf

D-MINT zielt auf die Entwicklung und Erprobung innovativer Technologien für das modellbasierte Testen ab, das in der Lage ist, verschiedenste Systemaspekte und Modelltypen zu kombinieren. Diesen neuen Ansatz nennen wir architekturgetriebenes modellbasiertes Testen.

D-MINT wird als Ergebnis modellbasierte Technologien bereitstellen, die in verschiedenen Fallstudien in der Industrie erprobt werden und dadurch in der Industrie direkt einsetzbar sind. Die mit den Fallstudien gesammelten Erfahrungen werden in Form von Demonstratoren veranschaulicht. Für jede Fallstudie wird ein Demonstrator aufgebaut, der in der Lage ist, den Einsatz von modellbasierten Testtechnologien an realen, Industrienahen Systemen zu veranschaulichen. Diese Demonstratoren werden auf verschiedenen Ausstellungen gezeigt und somit dazu beitragen, den Erfolg von D-MINT zu verbreiten.

## **2. VORAUSSETZUNGEN, UNTER DENEN DAS VORHABEN DURCHGEFÜHRT WURDE**

Zum Start des Projekts gab es eine Reihe model-basierter Testansätze. Die Verbreitung und Anwendung dieser Technologien in der industriellen Praxis war jedoch gering und beschränkte sich meist auf spezifische Anwendungen innerhalb spezieller Industriedomänen. Dementsprechend war die grundsätzliche Expertise der Partner im Testbereich stark ausgeprägt und dennoch sehr unterschiedlich. Eine konsequente durchgängige kommerzielle Nutzung der modellbasierten Testtechniken war nicht vorhanden.

Das Projekt basierte insbesondere auf den folgenden methodischen und technischen Vorarbeiten der Partner:

- Funktionales Testen mit TTCN-3
- Testbeschreibung mit dem UML2 Test Profil
- Systemmodellierungen, z.B. mit UML oder Matlab/Simulink
- Statistisches Testen

Der Stand der Projektpartner wird im Folgenden beschrieben:

- FOKUS beschäftigt sich seit Jahren mit modellbasierten Tests und war maßgeblich an der Entwicklung von TTCN-3 beteiligt. Es wurden erste Ansätze zur Testableitung, Testsimulation, Testvalidierung, verteilten Testausführung und zum UML Testprofil erarbeitet, wobei diese Ansätze im Rahmen dieses Projekts zu verallgemeinern und für die speziellen Anwendungsbereiche zu erweitern sind. FOKUS kann auf mehrjährige Erfahrungen beim Testen, der Spezifikation und der Leistungsanalyse von verteilten Systemen verweisen. Die Arbeitsgruppe MOTION beschäftigt sich schwerpunktmäßig mit Konformitäts-, Interoperabilitäts- und Leistungstests von Telekommunikationssystemen, -diensten und -komponenten, von Middleware Plattformen und Plattform-basierten Systemen. Grundlage sind verschiedene Vorarbeiten zur Weiterentwicklung von TTCN, UML, verteilte Testarchitekturen, Testmuster, TTCN-3/CTE.
- Nokia Siemens Networks gehört weltweit zu den führenden Unternehmen der mobilen Kommunikationsindustrie. Mit seinen Erfahrungen, seinen innovativen Produkten sowie bedienerfreundlichen und sicheren Lösungen ist das Unternehmen der führende Anbieter von Kommunikationsdiensten. Ein Teil seiner Führerschaft basiert auf der Entwicklung und Nutzung aktuellster Werkzeuge und Technologien. Bei den Entwicklungen wurde auch TTCN-3 für klassisches Konformitätstesten eingesetzt und modellbasierte Techniken für die Entwicklung von Telekommunikationssoftware verwendet.
- PikeTec ist auf die Qualitätssicherung für eingebettete Regelungs- und Steuerungssysteme spezialisiert und bietet in diesem Segment umfassende Beratungsleistungen und innovative Werkzeuglösungen für die Verbesserung der Software-entwicklung. Die fachlichen Ziele von PikeTec sind die Reifegradverbesserung der Entwicklungsprozesse und die Optimierung der Qualität der entstehenden eingebetteten Produkte. Ein Ergebnis dieser Arbeiten ist die Entwicklung der Testmethode „TPT“ sowie des gleichnamigen Testwerkzeugs, das auf den Test eingebetteter Steuerungs- und Regelungssysteme spezialisiert und in die modellbasierten Entwicklungsprozesse im Automobilbereich integriert ist und unter anderem den modellbasierten Test von MATLAB/Simulinkoder TargetlinkModellen unterstützt.
- Daimler: Auf dem Gebiet der modellbasierten Entwicklung besitzt Daimler bereits Erfahrungen. So werden verschiedene UML- und Simulink/StateFlow-Modelle in der Entwicklung bereits eingesetzt. Auch beim modellbasierten Test gibt es bereits einzelne Erfahrungen. Die bereits genannten Werkzeuge MTest und TPT (beide ursprünglich in der Daimler-Forschung entwickelt) ermöglichen den Test auf der Basis von Simulink/StateFlow-Modellen. Darüber hinaus gibt es Erfahrungen mit dem Test auf der Basis einzelner UML-Diagramme wie z.B. den Sequenz-Diagrammen (bzw. MSCs). Somit werden bei Daimler Modelle für die Entwicklung und auch für den Test von

softwarebasierter Fahrzeugelektronik teilweise bereits eingesetzt. Allerdings handelt es sich gerade beim Einsatz des modellbasierten Tests bisher noch um Insel-Lösungen, die größtenteils von der Daimler-Forschung getrieben werden. Was bisher fehlt, ist der durchgängige Einsatz dieser Technologie im Entwicklungsbereich von Daimler..

- Fraunhofer IESE: Eine der Kernkompetenzen des Fraunhofer Institut für Experimentelles Software Engineering liegt im Bereich Testen, wozu auch Erfahrungen in der Erzeugung von Testfällen für verschiedenartige Anwendungen, Anforderungen und Qualitätseigenschaften von Software gehören. Das Fraunhofer IESE ist aktiv an der Erforschung und Verbreitung modellbasierter Testtechnologien in industrieller Softwareentwicklung beteiligt. Es bestehen fundierte Kenntnisse u.a. zur Bewertung des Kosten/Nutzenverhältnisses von Testverfahren, der Entwicklung von risikobasierten Ansätzen zum Test nichtfunktionaler Qualitätseigenschaften und spezieller statistischer Testtechnologien.
- TestingTech hat sich auf die Entwicklung und Vermarktung von allgemeinen und speziellen Testwerkzeugen und Testlösungen spezialisiert. Das technologische Kernkonzept ist TTCN-3. Unter dem Markennamen TTworkbench bietet TestingTech eine hoch innovative Serie von Testwerkzeugen für TTCN-3 an, die den gesamten Zyklus der Testentwicklung und -ausführung unterstützt. Mit TTworkbench werden Testlösungen für ein breites Spektrum von industriellen Domänen entwickelt. TTworkbench unterstützt bereits heute standardmäßig ASN.1, IDL und XML. Damit öffnen sich weitere Möglichkeiten für modellbasiertes Testen, denn Daten aus Systemmodellen in diesen Sprachen können über TTworkbench für die Testspezifikationen in TTCN-3 verfügbar gemacht werden. In der aktuellen Weiterentwicklung von TTworkbench wird diese Arbeit fortgesetzt durch die Integration von UML2 und U2TP (UML 2.0 Testing Profile).
- ABB: Durch die ABB Energietechnik werden Strom-, Gas- und Wasserversorger wie auch Industrie- und kommerzielle Kunden betreut, mit einem breiten Spektrum an Produkten, Systemen und Dienstleistungen für Energieübertragung/Verteilung und Automatisierung. ABB Automatisierungstechnologien bieten eine Mischung aus zuverlässigen Produkten und einem Dienstleistungsportfolio mit Fachkompetenz für Endverbraucher und globaler Präsenz zur Bereitstellung von Lösungen für Steuerungsaufgaben, Triebwerkstechnik, Schutzeinrichtungen und Anlagenbau für die Prozessindustrie und Energieversorger. Zusätzlich zu den ABB Automatisierungslösungen für die Öl- und Gasindustrie entwickelt und liefert ABB Lummus Global Fertigungsstraßen, Raffinerien und petrochemische Anlagen. Die benutzten Testverfahren befinden sich auf einem hohen Automatisierungsniveau. MBT Ansätze wurden von ABB als sehr bedeutend eingeschätzt, jedoch bis Projektbeginn aber industriell noch nicht umgesetzt.
- Die iXtronics GmbH bietet Dienstleistungen und Entwicklungslösungen für den modellbasierten Entwurf mechatronischer Systeme an. Insbesondere das Werkzeug CAMeL-View TestRig, das sowohl in der Modellphase („virtueller Prototyp“), der Prüfstandsphase als auch in der Prototypenphase von industriellen Kunden eingesetzt wird, erlaubt den durchgängigen Entwurf mechatronischer Systeme von Anfang an. Hauptziele der iXtronics Produkte und Dienstleistungen sind die Reduktion der Entwicklungszeiten und -kosten sowie die frühzeitige Absicherung von Entwicklungsergebnissen.
- INSPIRE: Das Team der INSPIRE AG ist auf Eigenkapitalfinanzierungen, Projektverwertungen und Zukunftsprojekte spezialisiert. Das Team besitzt mehr als 15 Jahre Erfahrung in nationalen und internationalen Forschungsprojekten wie auch Projektreviews als Experten. Im Bereich europäischer Förderprojekte übernahm INSPIRE u.a. folgende Projekttrollen: Vorbereitung von Forschungsrahmenprogrammen, Evaluierung von Projektanträgen, Projektreviews als Experten, Forschung und Entwicklung, Arbeitspaket bzw. Innerhalb von D-MINT übernimmt INSPIRE keine wissenschaftlich-technischen Arbeiten, sondern die Markt und Wettbewerbsanalyse sowie die nationale und internationale Verwertungsstrategie.

 <b>D-MINT</b> <small>Deployment of Model-Based Technologies to Industrial Testing</small>	<b>D-MINT Abschlussbericht</b>	Seite : 8 of 81
		Version: 1.0 Datum: 30.06.2010
		Status : Final

### 3. PLANUNG UND ABLAUF DES VORHABENS

Das deutsche Teilprojekt von D-MINT war analog dem internationalen Projekt in 5 Arbeitspakete (AP) strukturiert:

- AP1: Fallstudien
- AP2: MBT Prinzipien und Methoden
- AP3: Werkzeuge
- AP4: Verwertung
- AP5: Verbreitung, Standardisierung und Training

Die Dauer des Projektes war geplant von 1.4.2007 bis 30.9.2009. Sie wurde im Laufe des Projektes aufwandsneutral bis zum 31.12.2009 verlängert, um sie an die Projektdauer der anderen europäischen Partner anzulegen, u.a. fanden das ITEA Symposium und die abschließende Projektbegutachtung im 4. Quartal 2009 statt. Ansonsten lief das Projekt entsprechend der Planung ab.

Organisatorisch wurden auf europäischer Ebene die Projektleitung (Colin Willcock, Nokia) und drei Work Package Leiter (Jens Herrmann (Daimler) für Fallstudien, Ina Schieferdecker (Fraunhofer FOKUS) für Testmethoden; und Thomas Bauer (Fraunhofer IESE) für Ergebnis-Verbreitung/Standardisierung) gestellt.

Darüber hinaus wurde vom deutschen Konsortium nicht nur die Fachveranstaltungsreihe MoTiP (Model-based Testing in Practice) initiiert, sondern auch eine Vielzahl von Workshops und Tutorien durchgeführt. Dies hat wesentlich zur Verbreitung der Ergebnisse beigetragen.

Das ITEA D-MINT Projekt erhielt den ITEA2 Symposium Exhibition Award 2009 für die beste Ausstellung auf dem genannten Symposium, das in Madrid veranstaltet wurde.

 <b>D-MINT</b> <small>Deployment of Model-Based Technologies to Industrial Testing</small>	<b>D-MINT Abschlussbericht</b>	Seite : 9 of 81 Version: 1.0 Datum: 30.06.2010 Status : Final
---	--------------------------------	--

## 4. WISSENSCHAFTLICHER UND TECHNISCHER AUSGANGSSTAND

Modellbasiertes Testen bezeichnet das Testen von Software, bei dem die Testfälle teilweise oder vollständig aus einem Modell abgeleitet werden. Das Modell beschreibt ausgewählte, meist strukturelle und funktionale, manchmal auch nicht-funktionale Aspekte des zu testenden Systems. Zusätzlich können die Tests selber modelliert, analysiert, validiert und in Bezug auf ihre Fehlererkennungsmöglichkeiten, Systemabdeckung u.a. quantifiziert werden.

Aktuelle modellbasierte Testansätze sind noch nicht leistungsfähig genug um, in industrieller Praxis in voller Breite eingesetzt zu werden. Darüber hinaus sind sie oftmals auf bestimmte Aspekte der Systemmodellierung beschränkt, z.B. Verhalten, Struktur oder Nutzen von Systemen und Komponenten, oder bedienen sich spezifischer Modellierungssprachen, z.B. UML, MATLAB Simulink.

Für den Projektverlauf war von den im Folgenden kurz umschriebenen drei Themenbereichen auszugehen:

### Modelbasiertes Testen

Nach Gartner Group [A2] sind modellbasierte Testmethoden auf Basis zugrundeliegender Modelle sowie Methoden zur Generierung und Auswertung der Testfälle zu unterscheiden. Die Auswahl von Generierungsmethoden ist in der Regel eng mit der Auswahl zweckmäßiger Modelle und Modellierungssprachen verbunden. Die Generierung von Testfällen kann automatisch oder manuell erfolgen.

Der manuelle Entwurf einzelner Testfälle erfolgt auf Basis von Heuristiken, z.B. Überdeckungsmaße, Äquivalenzklassenanalyse [A4][A16], aus dem formalen oder informalen Testmodell.

Die automatische Testfallgenerierung erfolgt in der Regel auf Basis graphentheoretischer Konzepte sowie analytischer und heuristischer Algorithmen aus einem formalen Testmodell [A1][A3]. Neben der problemspezifischen Implementierung dieser Algorithmen mittels imperativer Programmiersprachen werden zunehmend auch generische Ansätze mit deskriptiven Sprachen verwendet, z.B. Testfallgenerierung mittels Model Checking [A17].

### Modellgetriebene Softwareentwicklung

Mit der raschen Verbreitung strukturierter Modellierungstechniken [A8][A9] sowie deren Erweiterung zur Verhaltenssimulation und Echtzeitmodellierung [A10] in Verbindung mit einer breiten Unterstützung kommerziell verfügbarer Entwicklungswerkzeuge wurden formale und systematische Testansätze (z.B. bieten [A4] und [A16] einen guten Überblick) adaptiert und zu modernen modellbasierten Testmethoden erweitert, z.B. siehe [A3]. Durch die Entwicklung objektorientierter modellbasierter Softwareentwicklungsmethoden, z.B. UML-RT [A19][A5], SDL [A18], ROOM [A6], in Verbindung mit guter Werkzeugunterstützung, z.B. Rational Technical Developer [A24], iLogix Rhapsody [A23], Telelogic TAU [A25], wurde dieser Trend noch zusätzlich verstärkt. Die Model Driven Architecture [A21] der Object Management Group stellt einen nützlichen Ansatz zur Vereinheitlichung des Modellierungsprozesses dar.

Parallel zu softwaretechnischen Lösungen wurden aus ingenieurtechnischen Konzepten spezielle Methoden und Werkzeuge für den Entwurf signalverarbeitender Systeme entwickelt. Die Matlab Simulink Werkzeugkette von The Mathworks [A10] ist vermutlich einer der bekanntesten Vertreter dieser Entwicklungsansätze, andere sind z.B. Labview [A28], Modelica [A26], ASCET SD [A22] oder CAMeLView TestRig [A29].

Moderne Softwareentwicklungsmethoden, z.B. Catalysis [A11], Rational Unified Process [A12] oder die KobrA Methode [A13], zielen auf die durchgängige Prozessdefinition durch einheitliche Prinzipien und Techniken im gesamten Softwarelebenszyklus.

 Deployment of Model-Based Technologies to Industrial Testing	<b>D-MINT Abschlussbericht</b>	Seite : 10 of 81 <hr/> Version: 1.0 Datum: 30.06.2010 <hr/> Status : Final
--	--------------------------------	---

Im Bereich der modellbasierten Softwareentwicklung sind ebenfalls diverse Aktivitäten zur Vereinheitlichung bekannt. Modellbasierte Prozessmuster [A12], Notationen und Technologien, z.B. SysML [A21], UML [A19], TTCN3 [A14], U2TP [A15], sowie industrielle Standards, z.B. von IEC, DIN EN, FAKRA, ITU, und ISO werden zu Integrationszwecken überprüft und gegebenenfalls berücksichtigt.

### Quantifizierung von Qualitätseigenschaften

Diverse Arbeiten und Projekte haben bereits die Quantifizierung nicht-funktionaler Eigenschaften isoliert betrachteter Komponenten fokussiert, z.B. [A7][A34]. Eine Anzahl Projekte zielt auch auf die Erweiterung und Adaption dieser Methoden auf moderne Softwareentwicklungsansätze [A31][A32][A33]. Keines dieser Projekte konnte bisher eine praxistaugliche Lösung für die Quantifizierung von Qualitätseigenschaften großer verteilter Systeme liefern, z.B. für Steuerungssoftware im Automobil oder im Zug. Die Anwendung modellbasierter Technologien in einem nahtlosen und definierten Prozess erlaubt die durchgängige Verwendung statistischer Testmethoden zur Bewertung, Kontrolle und Optimierung der Produktqualität. Mit Cleanroom steht eine bewährte Methode zur Verfügung, welche rigoros Gebrauch von statistischen Testmethoden macht, allerdings nicht die notwendige Werkzeugunterstützung für einen breiten Industrieeinsatz besitzt [A7]. Durch die hohe industrielle Akzeptanz moderner modellbasierter Entwicklungsmethoden und -werkzeuge, z.B. Matlab/Simulink [A28] oder UML 2.0 [A19].

 Deployment of Model-Based Technologies to Industrial Testing	<b>D-MINT Abschlussbericht</b>	Seite : 11 of 81 <hr/> Version: 1.0 Datum: 30.06.2010 <hr/> Status : Final
--	--------------------------------	---

## 5. ZUSAMMENARBEIT MIT ANDEREN STELLEN

Der deutsche Anteil von D-MINT bestand aus der Zusammenarbeit der Partner FOKUS, Nokia Siemens Networks, Daimler, IESE, TestingTechnologies, ABB, iXtronics und INSPIRE. Dieser Anteil war zusammen mit finnischen, französischen, spanischen, estländischen und norwegischen Partnern in das europäische ITEA Projekt D-MINT eingebettet. Mit den Partnern auf europäischer Ebene wurde zu weiteren Fallstudien und zu Themen der Testmodellierung, Testgenerierung, der Testausführung und der Testbewertung zusammengearbeitet.

Zudem wurden die Ergebnisse des Projekts bei Standardisierungsgremien vorgestellt und in Standards eingearbeitet:

- ETSI (European Telecommunication Standards Institute) zu verschiedenen Aspekten von Testentwurf und Testbeschreibung mit Hilfe von Modellierungssprachen auf höherem Abstraktionsebenen
- ISTQB (International Software Testing Qualification Board) zur Entwicklung eines TTCN-3 Syllabus im Rahmen der Certified Tester Ausbildung auf Experten-Ebene

## Teil II: Projektergebnisse der Partner

### 6. FRAUNHOFER FOKUS, FÖRDERKENNZEICHEN 01 IS 07 001 A

#### 6.1 ERREICHTE ERGEBNISSE

Fraunhofer FOKUS war Technologiepartner im Projekt und hat organisatorisch das deutsche Teilprojekt von D-MINT und die Entwicklung der Testmethoden (AP2) koordiniert. Zudem hat es bei der Entwicklung der Lösungen und Werkzeuge in AP3 beigetragen. Dabei sind die folgenden wesentlichen Ergebnisse erarbeitet worden:

- Entwicklung einer Methodik und Implementierung zur Architektur-basierten Testentwicklung
- Entwicklung von Qualitätskriterien zur Testsuite Bewertung  
Implementierung eines Werkzeugs Analyse von TTCN-3 Testsuites
- Entwicklung einer Methodik, einer Beschreibungsnotation und Implementierung zur musterbasierten Testentwicklung
- Entwicklung einer Methodik und einer Implementierung zur Testgenerierung für eingebettete Systeme

#### 6.1.1 Thematische Ergebnisse

Die thematischen Ergebnisse werden im Folgenden dargestellt.

##### 6.1.1.1 Architektur-basierte Testableitung

Die Architektur-basierte Testableitung ([19], [36], [40]) setzt voraus, dass das zu testende System (SUT) in seiner Architektur möglichst umfassend definiert wurde. Idealerweise wird die SUT durch mehrere sich ergänzende Architekturbeschreibungen aus verschiedenen Sichten (Perspektiven/Aspekten) definiert und dargestellt. Die Form der Beschreibung wird nicht vorgegeben und kann sich an der jeweiligen Architekturperspektive orientieren. In unserem Fallbeispiel nehmen wir als Ausgangspunkt die von Ringler [5] eingeführten Architekturschichten von Systemmodellen (siehe Bild 6.1). Dabei wird ein System aus vier verschiedenen Perspektiven beschrieben: funktionale Anforderungen, logischer Systemaufbau, technische Realisierung, topologische Verteilung.

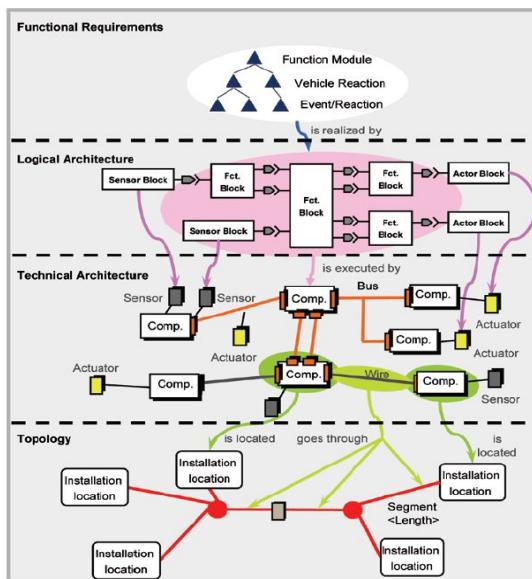


Bild 6.1: Eine Hierarchie von Architektursichten (aus [A35])

Die Elemente stehen in einer Architektschicht miteinander in Verbindung. Darüber hinaus gibt es auch Zusammenhänge zwischen den Architekturelementen verschiedener Perspektiven wie die Zuordnung funktionaler Anforderungen auf logische Blöcke, die Realisierung logischer Blöcke in technischen Komponenten oder die räumliche Verteilung von technischen Komponenten auf räumliche Beziehungen. Die Testbeschreibungen ergeben sich z.B. aus der Zusammenstellung aller Architekturinformationen, die sich auf eine Anforderung der funktionalen Perspektive beziehen. Sie beinhalten Informationen über Testkonfigurationen (die Test-Komponenten und ihre Ports/Schnittstellen) und Testabläufe, d.h. Vorbedingungen (preconditions), Testauslöser (stimulus events) sowie zu erwartende Beobachtungen (reactions), die auch als PERs (precondition-event-reaction Tripel) bezeichnet werden.

Bei der Architektur-basierten Testableitung werden insbesondere die über eine Systemsicht hinausgehenden Beziehungen für die Gewinnung von Testfällen benutzt. Algorithmen für die Testableitung können von jedem Element auf einer beliebigen Architektschicht beginnen und aufgrund der vorgegebenen Relationen die Informationen aus benachbarten Architektschichten mit einbeziehen. Beispiele sind: (1) Nach Auswahl eines Funktionsblocks aus der logischen Architekturperspektive können die zugehörigen Anforderungen auf der funktionalen Schicht ermittelt werden. (2) Nach Auswahl einer Funktion aus der funktionalen Requirementsicht werden die zugehörigen funktionalen Blöcke auf der logischen Schicht ermittelt.

Jede funktionale Anforderung kann sich auf mehrere logische Funktionsblöcke beziehen. Diese dienen nunmehr als Ausgangspunkt für die weitere Testableitung. Die architekturellen Verbindungen (Ports) dieser Funktionsblöcke werden in die Ebene der funktionalen Anforderungen mit einbezogen und erweitern damit das Architekturmodell auf dieser Schicht. Das gleiche Prinzip kann bei der Einbeziehung der technischen und topologischen Architekturperspektiven angewandt werden.

Neue Testfälle entstehen auch durch Identifizierung und Kombination von zusätzlichen Testdaten (z.B. unzulässige oder unbekannte Werte) sowie der sequentiellen oder parallelen Verknüpfung bereits vorliegender Tests. Weitere Testgenerierungsstrategien ergeben sich durch Umgestaltungen (z.B. Permutation von Preconditions) unter Berücksichtigung der evtl. im Modell beschriebenen Einschränkungen.

Im Rahmen einer Fallstudie wurden die mit dem vorgegebenen Systemmodell eingebrachten Informationen für die Erzeugung von Testbeschreibungen der funktionalen Anforderungen sowie der aus der logischen Architekturperspektive ermittelten Störfaktoren zusammengestellt. Die ursprünglich 28 zugrundegelegten Funktionen (PERs) könnten durch die Einbeziehung und Kombination von Störgrößen auf ca. 1700 Tests ausgeweitet werden, wobei diese Anzahl durch Techniken wie z.B. Datenpartitionierung oder Priorisierung sinnvoll zu begrenzt ist.

Da die automatische Ausführung der Testfälle die spezielle Syntax der eingesetzten Testausführungswerzeuge berücksichtigen müsste, wurde zunächst ein Hersteller unabhängiges tabellarisches Beschreibungsformat gewählt (s. Bild 6.2). Es enthält die aus dem PREEvision Systemmodell abgeleitete Testkonfiguration, Schnittstellentypen, Testabläufe sowie Testdaten.

**Test Configuration:**

**Following Test Components are needed:**

Spiegelauswahlschalter, Spiegelverstellschalter, Funktionen Zündschloss,

**The SUT is build by the following Components:**

Spiegelglasverstellung, Spiegel Motor Ansteuerung Fahrer, Schalter Erfassung

Spiegelverstellung.

**The SUT is to be stimulated through the following ports:**

Port (name)	Portprototype (name)
def16	Schalterstatus
def17	Spiegelverstellung
def29	Klemmenstatus

**Reactions are to be observed at the following ports:**

Port (name)	Portprototype (name)
def24	Motoransteuerung
def21	Motoransteuerung

**Additional ports at the SUT components, that can be used to inject disturbing signals:**

SUT Component Name	Port Name	Port Prototype Name
Spiegelglasverstellung	def20	Spiegelversstellung_Beifahrer_EQ
Spiegelglasverstellung	def13	Gang

**Test Behavior:**

### Bild 6.2: Testbeschreibung (Ausschnitt)

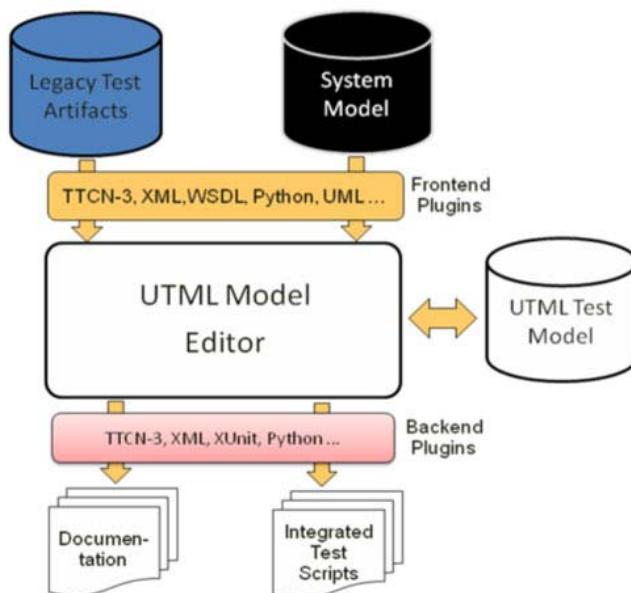
Für die Testausführung im Rahmen verschiedener Hardware Demonstratoren müssen die Testskripte in die jeweilige Testnotation der genutzten Testumgebung überführt werden. Bei Fraunhofer FOKUS wurde eine Anbindung und Ausführung mit einem TTCN-3 basierten Testsystem realisiert. Die erarbeitete Methode und Realisierung wurde im Rahmen des ITEA Symposiums und des D-MINT Abschlußreviews demonstriert. Nach Projektabschluß befindet sich ein Demonstrator im Automotivelabor von FOKUS und wird zur Präsentation der Forschungsergebnisse im Rahmen verschiedenster Veranstaltungen für Fachkollegen und Gäste vorgestellt. Die Demonstration kann unter Einbeziehung einer vollständigen Autotür oder auch nur mit einer einzelnen ECU Box gezeigt werden.

### Bild 6.3: ADT Generator Demonstrator

Für die Architektur-basierte Testentwicklung wurden darüber hinaus auch Ansätze zur Testpriorisierung erarbeitet [39]. Hierbei handelt es sich um Modelle die mit Hilfe von Zuweisungen von Prioritäten, Gewichten und Potentiale um zusätzliche Bewertungen erweitert wurden. Diese dienen der Optimierung der Testdurchführung in Hinblick auf eine den Prioritäten folgende Testausführungsreihenfolge.

### 6.1.1.2 Test Pattern-basierte Entwicklung von Test Suiten

Zur effizienten Entwicklung von Testsuiten und zur Erhöhung der Wiederverwendung von Testartefakten wurden ein UML-MOF Metamodel-Konzept erarbeitet und ein entsprechender Editor implementiert. Die Methode erlaubt die völlige Neuentwicklung von Testmodellen als auch die Einbeziehung von vorhandenen Testartefakten und Ansätze zur Definition und Nutzung von Testmustern in TTCN-3 sowie weiteren Beschreibungsnotationen (WSDL, XML etc.).



**Bild 6.4: Pattern-basierte Synthese von Testsuite unter Einsatz von UTML**

FOKUS entwickelte den sog. MDTester, ein Tool zur Unterstützung des modellgetriebenen Testentwicklungsprozesses, das es ermöglicht, Tests nach dem Ansatz der modellgetriebenen Softwareentwicklung zu produzieren

Die Implementierung wurde als ein Eclipse Plugin realisiert so dass eine Integration in die existierende Test- und Entwicklungsinfrastruktur vereinfacht wird. Der MDTester wurde erfolgreich bei der Entwicklung von TTCN-3 Test Suiten im Rahmen einer IMS Fallstudie genutzt. Weitere Details wurden veröffentlicht in [4], [20], [27], [34], [37] sowie [38]. Das Werkzeug sowie weitere Anleitungen können bei Fraunhofer FOKUS frei bezogen werden unter der Adresse:

[http://www.fokus.fraunhofer.de/en/motion/ueber\\_motion/technologien/utml/index.html](http://www.fokus.fraunhofer.de/en/motion/ueber_motion/technologien/utml/index.html).

### 6.1.1.3 Testqualitätsanalyse

Um für die Beurteilung einer Testsuite eine Qualitätsaussage treffen zu können ist zunächst ein Testqualitätsmodell zusammengestellt worden. Dieses Modell ist angelehnt an das allgemeine Softwarequalitätsmodell das in ISO 9126 standardisiert wurde.

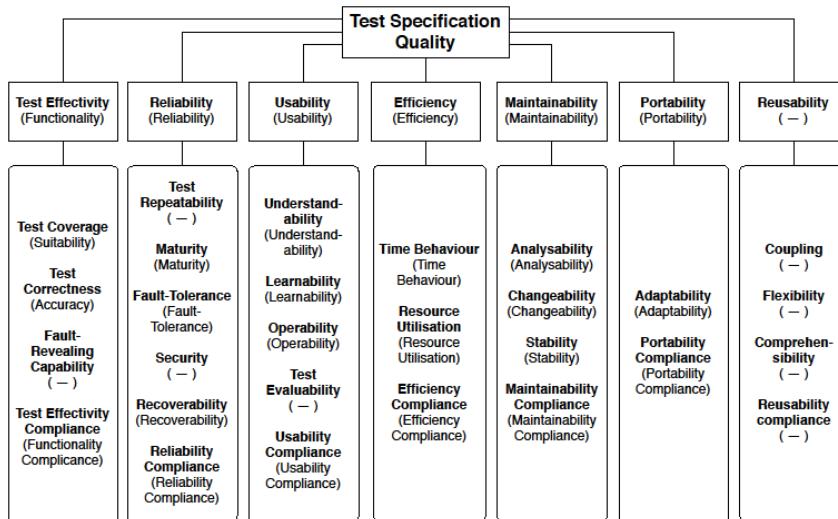


Bild 6.5: Test Qualitätsmodell

Auf der Grundlage von TTCN-3 Testsuites wurden Algorithmen erarbeitet und in einem Eclipse-plugin realisiert das Aussagen zur Abdeckung von Testdaten an Interface eines Testsystem berechnet. Desweiteren ermöglicht das Werkzeug eine Überprüfung zur Einhaltung von sog. TTCN-3 style Richtlinien. Das Werkzeug ist im Rahmen einer D-MINT Fallstudie mit Projektpartnern aus Estland eingesetzt worden in der TTCN-3 code aus einem MBT Werkzeug generiert wurde.

Language Element	Naming Convention	Prefix	Example	Statistics
Module	Use upper-case initial letter	none	IPv6Templates	0 / 47
Group within a node	Use lower-case initial letter	none	messageGroup	262 / 343
Data type	Use upper-case initial letter	none	SetupContents	0 / 264
Message template	Use lower-case initial letter	m_	m_setupInit	0 / 0
Message template with ...	Use lower-case initial letter	mw_	mw_anyUserReply	0 / 39
Signature template	Use lower-case initial letter	s_	s_callSignature	0 / 0
Port instance	Use lower-case initial letter	none	signallingPort	0 / 18
Test component instance	Use lower-case initial letter	none	userTerminal	0 / 0
Constant	Use lower-case initial letter	c_	c_maxRetransmission	0 / 362
External constant	Use lower-case initial letter	cx_	cx_necid	0 / 0
Function	Use lower-case initial letter	f_	f_authentication()	0 / 562
External Function	Use lower-case initial letter	fx_	fx_calculateLength()	0 / 4
Altstep	Use lower-case initial letter	a_	a_receiveSetup()	0 / 6
Test case	Use ETSI numbering	TC_	TC_CCR_0009_47_ND	0 / 268
Variable (local)	Use lower-case initial letter	v_	v_macId	5 / 78
Variable (defined within ...)	Use lower-case initial letter	vc_	vc_systemName	3 / 4
Timer (local)	Use lower-case initial letter	t_	t_wait	0 / 0
Timer (defined within a ...)	Use lower-case initial letter	tc_	tc_authMin	0 / 15
Module parameters	Use all upper case letters	none	PX_MAC_ID	0 / 99
Formal parameters	Use lower-case initial letter	p_	p_macId	0 / 97
Enumerated Values	Use lower-case initial letter	e_	e_syncOk	0 / 12

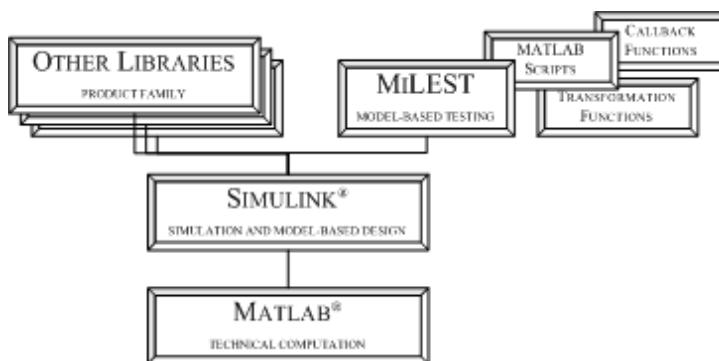
Bild 6.6: Testsuite Qualitätsüberprüfung

Die Details zu Algorithmen und Implementierungen wurden auf verschiedenen Konferenzen veröffentlicht, siehe [5], [25], [28] und [29].

### **6.1.1.4 Model-basiserte Testableitung für eingebettete Systeme**

Mit MiLEST entwickelte FOKUS eine neue Methode für die Stimulation und Bewertung des Verhaltens eingebetteter Hybridsysteme, die die Anforderungen in Eigenschaften spezifischer Signalmerkmale einteilt. Es wird ein neuartiges Signalverständnis definiert, das es erlaubt, eine abstrakte, auf der Basis der Signaleigenschaften beruhende Beschreibung zu erstellen – z. B. Einstieg, Konstante, Maximum.

Aus technischer Sicht ist MiLEST ein Simulink Add-on, das auf eine MATLAB Engine aufgesetzt wird, die ihrerseits eine Erweiterung für die modellbasierten Testaktivitäten bildet. MiLEST besteht aus einer Bibliothek, die Callback-Funktionen, Transformationsfunktionen und andere Skripte enthält. Die Test-Bibliothek ist in vier sich ergänzende Teile unterteilt.



**Bild 6.7: Technische Architektur der MiLEST Implementierung**

Folgende Beiträge wurden zu diesem Thema veröffentlicht: [11], [16], [21], [22], [24], [32], [33] sowie [42].

### **6.1.1.5 Verbreitung der Ergebnisse**

Neben den Veröffentlichungen zu verschiedenen Konferenzen hat FOKUS daran gearbeitet, die Projektergebnisse über weitere und neue Wege zu verbreiten.

Dazu hat FOKUS die Workshop-Reihe MoTiP mit initiiert und gemeinsam mit Projektpartnern erstmals im Sommer 2008 durchgeführt. Diese – und folgende Workshops – sind unter [http://www.fokus.fraunhofer.de/en/motion/news\\_events/veranstaltungsreihen/motip/index.html](http://www.fokus.fraunhofer.de/en/motion/news_events/veranstaltungsreihen/motip/index.html) zu finden.

Zudem war FOKUS aktiv bei der Gründung der ETSI Working Group on Patterns in Test Development und trägt mit seinen Arbeiten zur Definition von Test Pattern/UMTL in TTCN-3 bei.

Des Weiteren hat FOKUS Tutorials und/oder Tracks zu MBT bei verschiedenen internationalen Konferenzen beigetragen, z.B. [2], [7], [14], [17], [30] und [35].

## **6.1.2 Organisatorische Arbeiten**

Durch den ZE wurde die Projektleitung des deutschen Teilprojektes von D-MINT geleistet. Dazu waren viermal jährlich Treffen der deutschen Projektpartner vorzubereiten und durchzuführen. Diese dienten zur Information über den Projektstand, zur Koordinierung der Arbeiten innerhalb des deutschen und des europäischen Konsortiums und zur Vorbereitung der ITEA Projekttreffen.

Auf ITEA Ebene war der ZE für das WP2 zur Erarbeitung und Zusammenstellung von Model-basierten Testentwicklungsansätzen zuständig. Zudem wurden die halbjährlichen WP-Berichte an ITEA erstellt.

## 6.2 VORAUSSICHTLICHER NUTZEN

Der Nutzen für die beteiligte Arbeitsgruppe besteht bereits durch die grundsätzliche Stärkung und Verbreitung von Einsatzgebieten und Anwendungen der Modell-basierten Testentwicklung, da diese bei FOKUS einen Forschungsschwerpunkt einnimmt. Die Etablierung erlaubt unmittelbar eine Steigerung der Consulting- und Schulungsangebote. Entwickelte und aus der Projektarbeit noch abzuleitende Werkzeug-Prototypen sollen mit dem Ziel der Produktisierung und Lizenzierung an Softwarefirmen übertragen werden. Die erweiterte Werkzeugpalette der Testmodellierungs-Infrastruktur verbesserte wiederum sofort die Effizienz und führt zur Kostensenkung bei FOKUS-Testentwicklungen im Kundenauftrag.

Neben dem unmittelbar gewinnorientierten Nutzen bilden die aus der methodischen Projektarbeit entstandenen Arbeiten zu Dissertationen (z.B. MiLEST und UMTL) und die angrenzenden Forschungsarbeiten (u.a. Modelbus und einer Neufassung vom UTP) einen zukunftsweisenden wissenschaftlichen Fortschritt für FOKUS.

## 6.3 FORTSCHRITT BEI ANDEREN STELLEN

Während der Projektlaufzeit hat es neben den Ergebnissen der nationalen Projektpartner eine umfassende Verbreitung und Fortentwicklung der Model-basierten Testentwicklung gegeben. An dieser Stelle gilt es insbesondere die Ergebnisse der Projektpartner in Estland, Frankreich und Spanien zu benennen, da diese Beiträge durch die deutschen Arbeiten ergänzt wurden.

Die Vielfalt der durch D-MINT abgedeckten Industriedomänen zeigte sich auch in der Größe und dem Besucherinteresse auf dem ITEA Symposium 2009. Bei einem Rundgang auf dem Symposium wurde die Bedeutung der MBT Technologien darin erkennbar, dass sie in vielen weiteren ITEA eine große Rolle spielen.

Neben dem von D-MINT initiierten MoTiP Vortragsreihe entstanden im Projektzeitraum weitere neue Workshops, Konferenzen und Journals mit besonderer Berücksichtigung von model-basierter Testentwicklung. Die aus D-MINT gewonnene Expertise ermöglichte zahlreichen Projektmitarbeitern eine Beteiligung in den Progammausschüssen bzw. editoriellen Fachgremien.

Im und unmittelbar im Anschluss an den Projektzeitraum bestanden bei FOKUS Aktivitäten im Rahmen von Testentwicklungsprojekten die von den erarbeiteten neuen Werkzeugen Gebrauch machten. Hervorzuheben sind z.B. die Arbeiten mit TTCN-3 für industrielle Standardisierungsgremien.

## 6.4 ERFOLGTE ODER GEPLANTE VERÖFFENTLICHUNGEN

- [1] I. Schieferdecker , J. Großmann: Testing Embedded Control Systems with TTCN-3, An Overview on TTCN-3 Continuous, 5th IFIP Workshop on Software Technologies for Future Embedded and Ubiquitous Systems, Santorini, Greece, April 07.
- [2] I. Schieferdecker: Model-Based Testing, ASQF Seminar, Erlangen, Mai 07.
- [3] J. Grabowski, H. Neukirchen, I. Schieferdecker, D. Vega, B. Zeiss: An ISO 9126-based Quality Model to Assess the Quality of TTCN-3 Test Specifications. 4th TTCN-3 User Conference, Stockholm (S), Mai 07.
- [4] A. Vouffo-Feudjio, I. Schieferdecker: Pattern-based development of TTCN-3 test suites. 4th TTCN-3 User Conference, Stockholm (S), Mai 07.
- [5] D. Vega, I. Schieferdecker, G. Din: Test Data Variance as a Test Quality Measure: Exemplified for TTCN-3, Testing of Software and Communicating Systems, 19th IFIP TC6/WG6.1 International Conference, TestCom 2007, 7th International Workshop, FATES 2007, Tallinn, Estonia June 07.
- [6] J. Zander-Nowicka, A. Marrero Pérez, I. Schieferdecker: From Functional Requirements through Test Evaluation Design to Automatic Test Data Retrieval - a Concept for Testing of Software Dedicated for Hybrid Embedded Systems, The 2007 World Congress in Computer Science, Computer Engineering, & Applied Computing; The 2007 International Conference on Software Engineering Research and Practice, Las Vegas (USA), Juni 07.

- [7] I. Schieferdecker: Modellbasiertes Testen, OBJEKTspektrum, Heft 03/2007, SIGS-DATACOM, Juni 2007
- [8] J. Zander-Nowicka, A. Marrero Pérez, I. Schieferdecker, Z. R. Dai: Test Design Patterns for Embedded Systems, 10th International Conference on Quality Engineering in Software Technology, CONQUEST 2007, Potsdam (D) Sept. 2007.
- [9] J. Großmann , I. Schieferdecker: Testing Embedded Control Systems with TTCN-3, Enhancing TTCN-3 with Concepts for Continuous Streams. STTT Special Issue on TTCN-3 Evolution, Juni 07.
- [10] I. Schieferdecker: Test Generation as Model Transformation, TAROT Summer School, June 07.
- [11] J. Zander-Nowicka: Model Based Testing of Embedded Real-Time Systems, University of California in San Diego (U.S.A.), Computer Science and Engineering Department, Service-Oriented Software & Systems Engineering Laboratory, Sep. 07.
- [12] I. Schieferdecker: Test und Analyse sicherheitsgerichteter Systeme. Deutscher Kongress, Stuttgart, Oct 07.
- [13] I. Schieferdecker: Test Specifications as a Basis for common Test Requirements between OEM and Vendor. Systematic Testing Conference, Düsseldorf, Oct. 07.
- [14] I. Schieferdecker: Integrierte Systementwicklung und Test – Modelle als Grundlage effizienter Qualitätssicherung, MID Insight'07, Nov. 07
- [15] I. Schieferdecker: A test model quality framework. Model-Based Engineering of Embedded Real-Time Systems, Dagstuhl (D), Nov. 07.
- [16] J. Zander-Nowicka: Reactive Testing and Test Control of Hybrid Embedded Software. 5th Workshop on System Testing and Validation STV07, in conjunction with ICSSEA 2007. Paris (F) December 07
- [17] I. Schieferdecker: Paradigmenwechsel im Softwaretest, Fraunhofer IESE Akademie, Kaiserslautern, Feb. 08
- [18] A. Hoffmann, I. Schieferdecker: Modellgestütztes Testen komplexer Systeme. 5tes GI/ASQF/IRC, Schloß-Steinhöfel-Seminar, „Zukunft gestalten“, Steinhöfel (D), Apr. 08.
- [19] K.-D. Engel, A. Rennoch, I. Schieferdecker: Architecture-driven Test Development, MoTiP 08, Berlin, June 08.
- [20] A. Vouffou Feudjio: A Unified Approach to Test Modelling, MoTiP 08, Berlin, June 08.
- [21] J. Zander-Nowicka: Model-Based Testing of Real-Time Embedded Systems for Automotive Domain, Symposium on Quality of Embedded Systems, Berlin, Juni 08.
- [22] I. Schieferdecker: Testing Embedded systems, TAROT summer school, Bath, UK, Juni 08.
- [23] C. Hein, T. Ritter, M. Wagner: Model-Driven Tool Integration Framework – ModelBus. 4th European Conference on Model Driven Architecture Fondations and Applications (ECMDA 2008), Berlin, Juni 08.
- [24] Zander-Nowicka, Justyna; Schieferdecker, Ina: Systematic Test Data Generation for Embedded Software. SERP'08, The 2008 International Conference on Software Engineering Research and Practice, Las Vegas (USA), Jul 08.
- [25] Vega, Diana-Elena; Din, George; Schieferdecker, Ina: TTCN-3 Test Data Analyser using Constraint Programming. Nineteenth international Conference on Systems Engineering (ICSENG 2008), Las Vegas (USA), Aug 08.
- [26] Zander-Nowicka, Justyna; Rennoch, Axel; Bauer, T.; Eschbach, Dr. Robert: Modellbasierte Testentwicklung in der industriellen Anwendung, Objectspectrum “Testing”, SIGS Datacom, Oktober 08.
- [27] A. Vouffou: Quality of Test Specification by Application of Patterns. 2nd Workshop on Software Patterns and Quality (SPAQu'08), Nashville, Tennessee, USA, Oct. 08.
- [28] Vega, Diana-Elena; Din, George; Schieferdecker, Ina: Automated Maintainability of TTCN-3 Test Suites based on Guideline Checking: The 6th IFIP Workshop on Software Technologies for Future Embedded and Ubiquitous Systems (SEUS 2008), Capri (I), Oct 08.
- [29] Vega, Diana-Elena; Din, George; Schieferdecker, Ina; Arbanowski, Stefan: Application of Clustering Methods for Analysing of TTCN-3 Test Data Quality, The Third International Conference on Software Engineering Advances ICSEA 2008, Sliema (Malta), Oct. 08

 Deployment of Model-Based Technologies to Industrial Testing	<b>D-MINT Abschlussbericht</b>	Seite : 20 of 81 Version: 1.0 Datum: 30.06.2010 <hr/> Status : Final
---	--------------------------------	---

- [30] T. Bauer, A. Rennoch, I. Schieferdecker: Model-Based Testing and its use in the ITEA2 project D-MINT, ICSSEA 2008, 21st edition of the ICSSEA International Conference on Software & Systems Engineering and their Applications, Paris (F) Dez 08.
- [31] Vouffo, A., Kingni, A.: Model-Driven Engineering of IMS Tests. 6th Workshop on SYSTEM TESTING AND VALIDATION (STV08), Madrid (E), Dezember 08.
- [32] J. Zander-Nowicka: Model-based Testing of Embedded Systems in the Automotive Domain, TU Berlin, IRB Verlag, Januar 2009
- [33] J. Zander-Nowicka: Quality assurance of automotive hybrid embedded software. 5th IFIP Summer School & 2nd ACM/IEEE ICSE Workshop, Cape Town, SouthAfrica, March 09.
- [34] Alain Vouffo: Model-driven Functional Test Engineering for Service Centric Systems, TridentCom09, Washington DC, April 09.
- [35] T. Bauer, A. Rennoch: TTCN-3 and its role and usage in MBT from the D-MINT perspective, TTCN-3 user conference, Sophia Antipolis (F), June 09.
- [36] G. Din, K.-D. Engel: An Approach for Test Derivation from System Architecture Models applied to Embedded Systems, MoTiP workshop, Enschede (NL), June 09.
- [37] A. Vouffo, I. Schieferdecker: A Pattern Language of Test Modelling for Reactive Software Systems. Symposium at European conference on patterns (EuroPLoP2009), July 09.
- [38] A. Vouffo: Model-driven Functional Test Engineering for Service Centric Systems, Workshop on Visual formalisms for patterns, Oregon (USA) 24.9.2009.
- [39] A. Hoffmann, A. Rennoch, I. Schieferdecker, N. Radziwil: A Generic Approach for Modeling Test Case Priorities with Applications for Test Development and Execution. Modellbasiertes Testen (MoTes 2009), Lübeck (D) 29.09.2009.
- [40] G. Din, K. Engel, A. Rennoch: Architekturbasierte Testentwicklung für eingebettete Systeme, OBJEKTSPEKTRUM, Ausgabe Testing/2009, SIGS Datacom, Oktober 09.
- [41] A. Vouffo: Testmodellierungen in Industrie und Forschung - Überblick und Beispiele. ASQF Fachreferat. Berlin, Mai 2010.
- [42] J. Zander, I. Schieferdecker: Model-based Testing of Embedded Systems Exemplified for the Automotive Domain, Book chapter in In: Behavioral Modeling for Embedded Systems and Technologies: Applications for Design and Implementation, Editors: Gomes, L., Fernandes, J. M., Idea Group Inc (IGI) July 2010.
- [43] Ch. Buckl, A. Knoll, I. Schieferdecker, J. Zander: Model-Based Analysis and Development of Dependable Systems. Book chapter In: Model-Based Engineering of Embedded Real-Time Systems, LNCS State of the Art-Surveys for the MBEERTS Dagstuhl Seminar, 2010.
- [44] A. Rennoch, S. Lüdtke, K.D. Engel: Architecture-Driven Testing using TTCN-3. Submitted for ICSSEA'2010, Paris, Dec. 2010.

 Deployment of Model-Based Technologies to Industrial Testing	<b>D-MINT Abschlussbericht</b>	Seite : 21 of 81 Version: 1.0 Datum: 30.06.2010 <hr/> Status : Final
--	--------------------------------	---

## 7. NOKIA SIEMENS NETWORKS, FÖRDERKENNZEICHEN 01 IS 07 001 B

Nokia Siemens Networks als ZE hatte im Rahmen des Projektes die Eignung der entwickelten Methoden für modell-basiertes Testen zu überprüfen. Dabei hat der ZE zu der Definition der Fallstudie, der Definition der Anforderungen an die Werkzeuge und Methoden, und zu der Auswertung der Fallstudie beigetragen. Diese Arbeiten wurden zusammen mit Nokia Siemens Networks in Finnland im Rahmen von D-MINT durchgeführt. Zusätzlich zu den eigentlichen Projektaufgaben wurde vom ZE die Projektleitung des gesamten ITEA-Projektes D-MINT durchgeführt.

### 7.1 ERREICHTE ERGEBNISSE

Außer der thematischen Arbeit war der ZE auch für die Projektleitung des ITEA-Projektes verantwortlich. Die thematischen Ergebnisse sind in Abschnitt 7.1.1 und die Ergebnisse der Projektleitung in Abschnitt 7.1.2 dargestellt.

#### 7.1.1 Thematische Ergebnisse

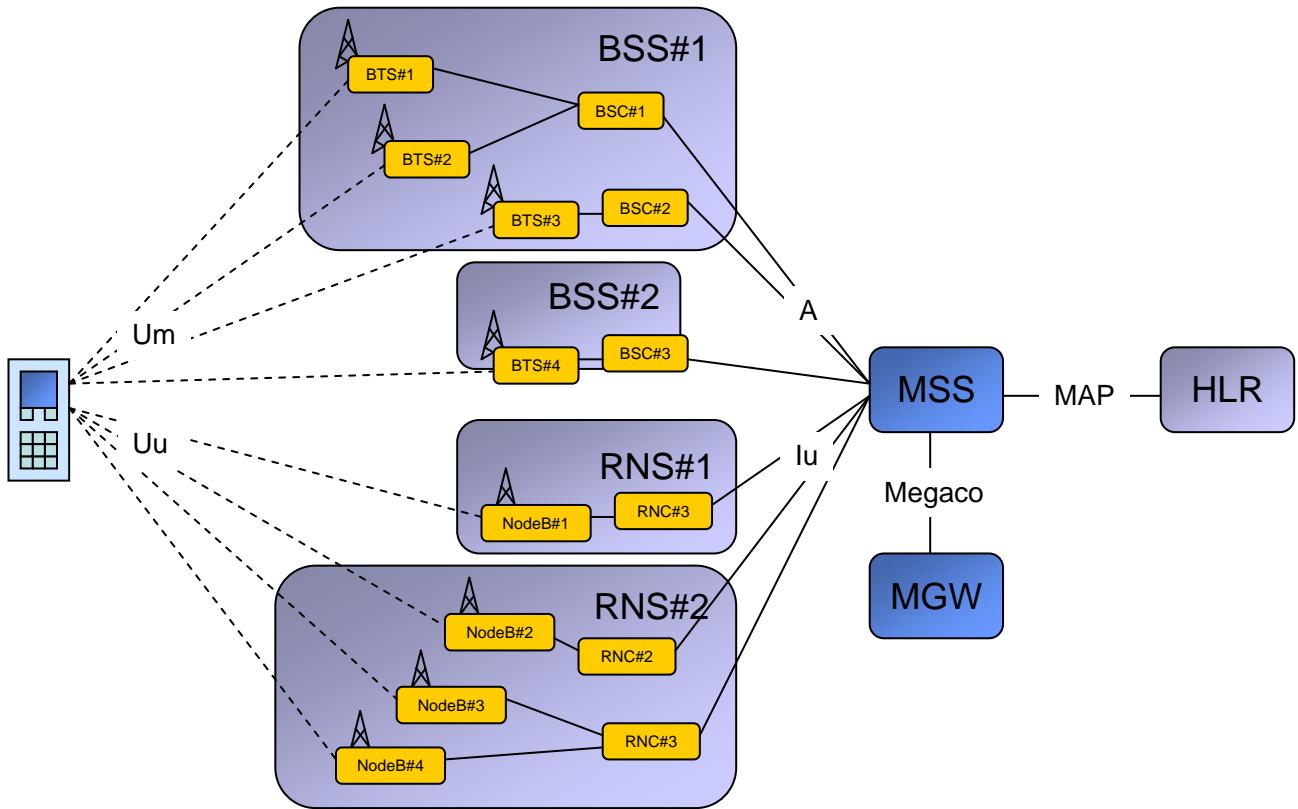
Der inhaltliche Beitrag des ZE war im Wesentlichen die Durchführung einer Fallstudie. Über die Fallstudie hinaus wurden Beiträge zu einer übergeordneten Methode zum modell-basierten Testen eingebetteter System geleistet.

Die Ergebnisse wurden in enger Zusammenarbeit mit Nokia Siemens Networks Finnland, ebenfalls gefördert im Rahmen des ITEA Projektes D-MINT, erzielt.

##### 7.1.1.1 *Definition der Fallstudie*

Als getestetes System (System Under Test, SUT) in der Fallstudie wurde ein Netzwerkelement aus dem Bereich der Mobilkommunikation ausgewählt. Das SUT ist ein Mobile Switching Server (MSS) aus Mobilfunknetzen der zweiten und dritten Generation. Es ist über standardisierte Schnittstellen mit Funknetzen der zweiten Generation (GSM-BSS) als auch mit Funknetzen der dritten Generation (UMTS-RNS) verbunden. Es ist mit dem Home Location Registry verbunden um Zugang zu den Benutzerdaten zu haben und es ist mit einem oder mehreren Media Gateway verbunden, das die eigentlichen Datenströme handhabt. Das MSS ist das wesentliche Netzwerkelement, das das Aufsetzen von Anrufen und die Übergabe von Benutzern zwischen verschiedenen Radionetzen oder Subnetzen in einem einzelnen Radionetz steuert.

Die Netzwerkarchitektur ist in Bild 7.1 dargestellt.



**Bild 7.1: Netzwerkarchitektur**

Das SUT ist ein typisches Netzwerkelement im Mobilfunkbereich. Erzielte Ergebnisse lassen sich deshalb auch auf andere Netzwerkelemente aus dem Anwendungsbereich übertragen. Das gewählte SUT wurde parallel zu der Projektdurchführung mit herkömmlichen Mitteln weiterentwickelt und getestet. Es bestand die Möglichkeit, die im Projekt erzeugten Tests mit denen aus der herkömmlichen Entwicklung zu vergleichen. Ebenfalls war es möglich, die erzeugten Tests auf das in der Entwicklung befindliche SUT anzuwenden.

Für die Durchführung der Tests muss das Testsystem die Umgebung des SUTs simulieren, genauer das Verhalten der Umgebung an den Schnittstellen des SUTs. Hierzu wurden der Protokollimulator EAST und das Testwerkzeug QTronic der beiden Projektpartner Nethawk bzw. Conformiq eingesetzt. EAST und QTronic waren auf handelsüblichen PCs oder Laptops installiert. Das SUT stand sowohl als echte Hardware als auch als PC Emulation zur Verfügung. In beiden Fällen konnte das SUT mit den Testwerkzeugen über IP verbunden werden.

Ziel der Fallstudie war es, die folgenden Fragen zu untersuchen:

- Lassen sich aus Systemmodellen überhaupt Testfälle generieren und ausführen?
- Kann der Benutzer die erzeugten Testfälle verstehen?
- Lässt sich die Anzahl der erzeugten Tests steuern?
- Wieviel Zeit wird benötigt die Tests zu erzeugen und auszuführen?
- Welche Eigenschaften hat das Systemmodell?
- Wie können derselbe oder andere Tests nach dem Auffinden eines Fehlers fortgesetzt werden?

Die Fallstudie ist genauer in [47] und [48] beschrieben.

### 7.1.1.2 Anforderungen an Methoden und Werkzeuge

Um das gewählte SUT modellbasiert zu testen, mussten die entsprechenden Modelle erstellt, validiert, und bei Bedarf geändert werden. Aus den Modellen waren Tests zu erzeugen, durchzuführen, und zu

analysieren. Das Ergebnis der Analyse konnte zu einer Änderung der Modelle oder auch zu einem Fehlerbericht über das SUT führen. Die verschiedenen Aufgaben wurden von mehreren Mitarbeitern an verschiedenen Standorten durchgeführt. Die Rahmenbedingungen für die Projektdurchführung waren damit zumindest ähnlich zu herkömmlichen Entwicklungsmethoden.

Aus den beschriebenen Abläufen und Rahmenbedingungen wurden Anforderungen an die Methoden und Werkzeuge für modell-basiertes Testen definiert. Über die in Abschnitt 7.1.1.1 angesprochenen Fragestellungen hinaus geht es hier darum mit den Methoden und Werkzeugen auch effizient arbeiten zu können.

Die Anforderungen wurden in verschiedene Kategorien klassifiziert

- Unterstützung durch CASE (Computer Aided Software Engineering) Werkzeuge
- Verfolgbarkeit von Anforderungen
- Art und Eigenschaften der Modelle
- Modellanalyse
- Echtzeitunterstützung in Modellen und Testausführung
- Benutzbarkeit
- Teststrategien und Testabdeckung
- Testausführung
- Testberichte

Die Anforderungen sind ausführlich in [48] beschrieben.

### **7.1.1.3 Durchführung und Auswertung der Fallstudie**

Die Fallstudie wurde zweimal durchgeführt, dabei war der Umfang der Modelle in beiden Durchläufen unterschiedlich. In beiden Durchführungen wurde bewertet wieweit die Methoden und Werkzeuge die zuvor aufgestellten Anforderungen erfüllen. Der detaillierte Aufbau des SUTs in der Testumgebung und die Bedienung der Testumgebung sind in [52] beschrieben, die beiden Auswertungen in [53] und [54].

Die Fallstudie hat gezeigt:

- Systemmodelle sind geeignet um Testfälle daraus automatisch zu generieren. Die generierten Tests können ebenfalls automatisch ausgeführt und ausgewertet werden.
- Durch die Ableitung aus den Systemmodellen lassen sich die erzeugten Tests auch wiederum als Folge von Ausführungsschritten in den Systemmodellen verstehen. Die erzeugten Tests sind daher für den Benutzer verständlich.
- Die Anzahl der Tests lässt sich durch verschiedene Überdeckungskriterien steuern. Die Überdeckungskriterien können sich dabei sowohl auf die Anforderungen als auch auf das Modell selbst, wie z.B. Kantenüberdeckung, beziehen.
- Die Erzeugung der Tests war ausreichend schnell um interaktives Arbeiten zu ermöglichen. Die erzeugten Tests konnten in Echtzeit ausgeführt werden, dabei ist zu beachten, dass die zeitlichen Abläufe im Anwendungsgebiet im Sekundenbereich liegen.
- Das Systemmodell beschreibt das Verhalten des Systems in Form von Automaten.
- Einzelne Tests können nach Erkennen eines Fehlers nicht sinnvoll fortgesetzt werden. Die Ausführung weiterer Tests ist problemlos möglich.

Durch die Fallstudie wurde gezeigt, dass die entwickelten systematischen Prozesse und die entwickelte Kette von Werkzeugen in der Praxis einsetzbar sind. Modell-basiertes Testen als Technologie stellt ausreichende Prozesse und Methoden für den industriellen Einsatz bereit. Systemmodelle sind geeignet als Ausgangspunkt modell-basierten Testens. Allerdings ist klar, dass die Einführung modell-basierten Testens im großen Maßstab einen entsprechend hohen Aufwand benötigen würde um die existierenden Testsysteme abzulösen.

Die Auswertung der Fallstudien hat gezeigt, dass Modellierungskompetenz relativ einfach aufzubauen ist. Modellierung an sich wird durch den Werkzeugunabhängigen Modellentwicklungsprozess unterstützt. Refaktorierung und Wartung der Modelle konnten in den Fallstudien nicht ausreichend berücksichtigt werden. Allerdings sind beide Aspekte stärker Werkzeugabhängig und müssen bei der Auswahl von Modellierungswerkzeugen bewertet werden.

Einige Probleme aus der herkömmlichen Testwerkzeugentwicklung wurden auch im Rahmen des modell-basierten Testens nicht gelöst. Als Beispiel ist hier die mangelnde Integration verschiedener Testwerkzeuge zu nennen. Jedes Werkzeug muss für sich mit anderen Werkzeugen integriert werden. Auch die Verwendung von UML in verschiedenen Werkzeugen brachte keine Vorteile bei der Integration. Die

 <b>D-MINT</b> <small>Deployment of Model-Based Technologies to Industrial Testing</small>	<b>D-MINT Abschlussbericht</b>	Seite : 24 of 81 Version: 1.0 Datum: 30.06.2010 <hr/> Status : Final
---	--------------------------------	---

verschiedenen UML Werkzeuge verwenden unterschiedliche Varianten von XMI (XML Metadata Interchange), so dass auch XMI nicht die Rolle als gemeinsames Dateiaustauschformat übernehmen kann.

#### **7.1.1.4 Methode zum modell-basierten Test eingebetteter Systeme**

Im Rahmen des ITEA Projektes D-MINT wurde eine übergeordnete Methode zum modell-basierten Test eingebetteter System definiert [57]. Der ZE hat zu dieser Methode die Instantiierung auf den Mobilkommunikationsbereich beigetragen.

Die wesentlichen Sichten sind die Architektursicht für funktionale Anforderungen, die logische und die technische Sicht. Anforderungen sind oft in natürlicher Sprache beschrieben. Die Tests werden automatisch aus Systemmodellen abgeleitet, die mit den Anforderungen verknüpft sind. Damit kann die Überdeckung der Anforderungen durch die Tests analysiert werden.

#### **7.1.2 D-MINT Projektleitung**

Der ZE hatte die Projektleitung des gesamten ITEA Projektes D-MINT inne. Im Verlauf des ITEA Projektes kamen neue Teilnehmer hinzu, der ZE hat entsprechend den FPP aktualisiert. Für das ITEA Projekt waren drei jährliche Treffen aller europäischen Projektpartner vorzubereiten und durchzuführen. Im Rahmen des Projektes fanden 7 dieser Treffen statt.

Der ZE hat 2 ITEA Begutachtungen vorbereitet und durchgeführt. Beide Begutachtungen haben zu überaus positiven Resultaten geführt.

Das ITEA Projekt hat an 3 ITEA Symposien teilgenommen und dabei jeweils mehrere Demonstrationen präsentiert. Die Symposien wurden ebenfalls durch den ZE vorbereitet. 2009 hat das ITEA Projekt den ‚Best Exhibition Award‘ des ITEA Symposiums erhalten.

Auch für das Jahr 2010 ist die Teilnahme am ITEA Symposium geplant. Dabei sollen Folge- und über das Ziel hinausgehende Ergebnisse vorgestellt werden.

Der ZE führte die Resourcen- und Kostenplanung durch. Diese wurden erschwert durch die unterschiedliche Genehmigung der Zuschüsse in den verschiedenen europäischen Ländern. Zwischen der Genehmigung in den ersten und letzten Ländern lag eine Zeitspanne von 18 Monaten. Der ZE war ebenfalls verantwortlich für die Erstellung der vierteljährlichen Fortschrittsberichte auf europäischer Ebene.

Trotz der Größe des Projektes – 25 Firmen aus 7 Staaten die 8 Anwendungsgebiete abdeckten – konnte das Projekt innerhalb des Kosten- und Zeitrahmens gehalten werden. Dabei wurden exzellente Ergebnisse erzielt.

#### **7.2 VORAUSSICHTLICHER NUTZEN**

Ausgehend von den Projektergebnissen wurde ein firmeninternes Projekt gestartet, das modell-basiertes Testen in der Praxis umsetzen wird. Hierbei sind die im Rahmen des Projektes etablierten Kontakte mit den Werkzeugherstellern, insbesondere Conformiq, essentiell.

#### **7.3 FORTSCHRITT BEI ANDEREN STELLEN**

Während der Durchführung des Projektes ist das Interesse insgesamt an modellbasiertem Testen gestiegen. Als mögliche Grundlage für modell-basiertes Testen für eingebettete Systeme wurde das UML Testing Profil für eingebettete System UMLes definiert.

 Deployment of Model-Based Technologies to Industrial Testing	<b>D-MINT Abschlussbericht</b>	Seite : 25 of 81 <hr/> Version: 1.0 Datum: 30.06.2010 <hr/> Status : Final
--	--------------------------------	---

## 7.4 ERFOLGTE ODER GEPLANTE VERÖFFENTLICHUNGEN

Die Projektergebnisse sind, soweit nicht vertraulich, unter <http://www.d-mint.org> verfügbar. Weitere Veröffentlichungen sind:

- [45] Fredrik Abbors, Tuomas Pääjärvi, Risto Teittinen, Dragos Truscan, Johan Lilius, A Semantic Transformation from UML Models to Input for the Qtronic Test Design Tool, Technical report, TUCS Technical Reports Series, Turku Center for Computer Science, Turku, Finland, 2009
- [46] Fredrik Abbors, Tuomas Pääjärvi, Risto Teittinen, Dragos Truscan and Johan Lilius. Transformational Support for Model-Based Testing – from UML to QML, 2nd Workshop on Model-based Testing in Practice (MoTiP 2009), Enschede, 2009
- [47] Thomas Deiß, Nokia Siemens Networks Demonstrator Description, D-MINT deliverable D1.1.2, 2008
- [48] Thomas Deiß, Teuvo Eloranta, Risto Teittinen, Nokia Siemens Networks Case Study and Initial Requirements, D-MINT deliverable D1.2.2, 2007
- [49] Thomas Deiß, Model-based testing at Nokia Siemens Networks, Seminar Universität Göttingen, 2007
- [50] Thomas Deiß, Trends in TTCN-3: Language and Usage, Software & Systems Quality Conference, Düsseldorf, 2008, [http://www.sqs-conferences.com/de/2008/program\\_08.htm](http://www.sqs-conferences.com/de/2008/program_08.htm)
- [51] Thomas Deiß, Jens Grabowski, Gyorgy Rethy, Ina Schieferdecker, The new TTCN-3 version 4.1.1, TTCN-3 Users Conference 2009, Sophia Antipolis, 2009, <http://www.ttcn3.org/TTCN3UC2009/program.htm>
- [52] Risto Teittinen, Nokia Siemens Networks demonstrator user manual, D-MINT deliverable D1\_3\_NSN, 2008.
- [53] Risto Teittinen, Nokia Siemens Networks case study evaluation report, D-MINT deliverable 1.3.D1\_4\_NSN, 2009
- [54] Risto Teittinen, Nokia Siemens Networks case study evaluation report, D-MINT deliverable D1\_5\_NSN, 2009
- [55] Risto Teittinen, ETSI-MTS, Technical Report on the Application of Model-Based Testing in the Telecoms Domain, planned for March 2010.
- [56] Colin Willock, D-MINT: Bringing Model-Based Testing to Industry, invited talk, 8th Intl. Conference on Software QA and testing for Embedded Systems, 2009?, Bilbao
- [57] D-MINT consortium: D-MINT white paper, Common approach to architecture driven testing, D-MINT deliverable CA\_WP\_v1.6, [http://www.d-mint.org/public/CommonApproach\\_WhitePaper\\_DMINT.pdf](http://www.d-mint.org/public/CommonApproach_WhitePaper_DMINT.pdf)

Colin Willcock war Mitglied im Programmkommittee der Workshop Serie MoTiP, Model-based Testing in Practice.

## 8. PIKETEC, FÖRDERKENNZEICHEN 01 IS 07 001 C

PikeTec ist spezialisiert auf modellbasierte Testverfahrens- und Testwerkzeugentwicklung für den Test eingebetteter Steuerungs- und Regelungssysteme. Die Arbeiten von PikeTec im Rahmen von D-MINT hatten vorrangig das Ziel, die zunehmende Bedeutung der Qualitätssicherung und hierbei vor allem des Test eingebetteter Steuerungs- und Regelungssysteme sowie den wachsenden Bedarf nach qualitativ hochwertigen Prozessen und Methoden aufzugreifen und ausgereifte, auf die Bedarfe der kommenden Jahre optimierte Testverfahren und Werkzeuge für diesen Markt zu entwickeln. Das Testwerkzeug TPT, das auf den Test eingebetteter Steuerungs- und Regelungssysteme spezialisiert ist, wurde im Rahmen von D-MINT um neue Komponenten erweitert, die sich aus den veränderten Randbedingungen aufgrund des Wechsels von der klassischen zur modellbasierten Softwareentwicklung ergeben.

Die Aufgaben von PikeTec in D-MINT konzentrierten sich auf methodische Erweiterungen der modellbasierten Testtechnologie TPT sowie auf die Unterstützung und Durchführung der Fallstudien von Daimler und ABB gemeinsam mit anderen Partnern in D-MINT. PikeTec war in D-MINT als Testwerkzeug- und Testmethodenentwickler vorrangig verantwortlich für die Adaption von Methoden und Werkzeugen für den modellbasierten Test von Steuerungs- und Regelungssystemen auf der Basis seines Testwerkzeugs TPT. Dies erfolgte in enger Abstimmung mit den Industrie-Fallstudien und anderen Partnern. Im Vordergrund der Arbeiten standen die Unterstützung der Durchgängigkeit der modellgetriebenen Testinfrastruktur von Model-in-the-Loop-Tests über Software-in-the-Loop-Tests bis hin zu Hardware-in-the-Loop-Tests sowie die Unterstützung von Applikationsparametern in den Testmodellen.

### 8.1 ERREICHTE ERGEBNISSE

PikeTec hat im Rahmen des ITEA-Projektes D-MINT die in den folgenden Unterabschnitten beschriebenen Ergebnisse erzielt.

#### 8.1.1 Durchgängigkeit von Tests von MiL bis HiL

Bei der Entwicklung von Steuerungs- und Regelungssystemen wird in zunehmendem Maße das Paradigma der modellbasierten Entwicklung angewendet. Dies bedeutet, dass Systeme bereits sehr frühzeitig auf Modellebene entwickelt, simuliert, validiert und getestet werden können, ohne dass hierfür bereits reale Hardware vorhanden sein muss. Die frühzeitige Absicherung der Funktion ist ein wesentlicher Erfolgsfaktor für modellbasierte Entwicklungsprozesse, da hierdurch die Entwicklungskosten und die Entwicklungszeit durch kostspielige Fehlerbehebung in späteren Phasen der Entwicklung entfällt.

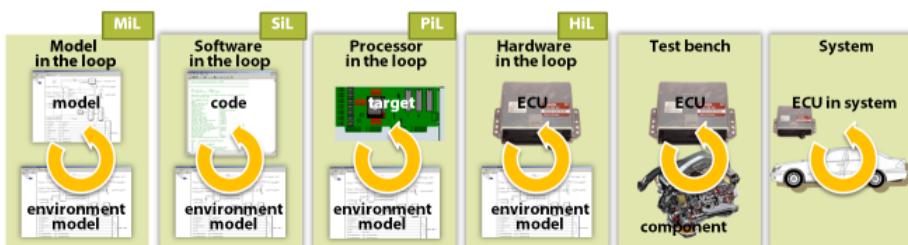
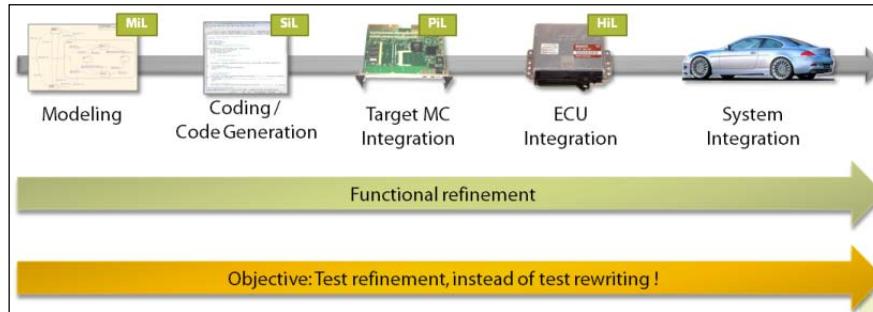


Bild 8.1 Integrationsebenen modellbasierter Entwicklung

PikeTec hat daher im Kontext von D-MINT untersucht, inwieweit sich modellbasierte Tests zwischen den Integrationsebenen der Entwicklung wiederverwenden lassen.



**Bild 8.2 Wiederverwendung von Tests von MiL bis HiL**

Bei dieser Wiederverwendung wurden folgende Schwerpunkte untersucht:

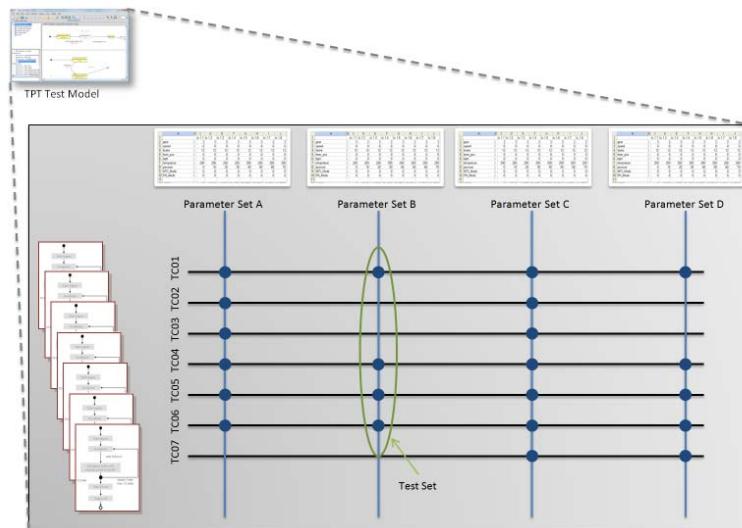
- Wiederverwendung der Testtechnologie: Identische Testmodellierungstechniken werden von MiL bis HiL verwendet.
- Wiederverwendung von Testmodellen mit Adaption der Schnittstellen: Testmodelle lassen sich 1:1 zwischen den Ebenen wiederverwenden. Die Unterschiede in den Schnittstellen werden durch Adapter zwischen Testmodell und zu testendem System sichergestellt.
- Wiederverwendung von Testauswertungen: System-Constraints und Erwartungswerte lassen sich zwischen den Ebenen wiederverwenden.

Im Rahmen der Daimler-Fallstudie wurden praktische Projektbeispiele betrachtet, bei denen die Durchgängigkeit im Kontext realer Entwicklungsprojekte mit allen drei Schwerpunkten exemplarisch erprobt wurde. Zu diesem Zweck wurde insbesondere eine prototypische Integration zwischen TPT und dem bei Daimler im Einsatz befindlichen HiL-System PROVEtech:TA entwickelt.

Auf dieser Grundlage konnte PikeTec im Rahmen von D-MINT nachweisen, dass die Durchgängigkeit von frühen Modelltests (MiL) bis zum Test im Hardware-Prüfstand (HiL) nicht nur theoretisch, sondern auch praktisch machbar ist. Außerdem wurde für diesen Ansatz identifiziert, unter welchen Randbedingungen welche positiven Effekte in Bezug auf die Entwicklungseffizienz und -effektivität und in Bezug auf die Qualität des Systems zu erwarten sind.

### 8.1.2 Unterstützung von Applikationsparametern in Testmodellen

Ein weiterer methodischer Schwerpunkt von PikeTec in D-MINT bestand in der Untersuchung von Möglichkeiten zur Integration von Applikationskonfigurationen und -parametern in die TPT-Testmodelle. Ziel war es, die Testmodelle möglichst unabhängig von konkreten Applikationskonfigurationen von Systemen definieren zu können, so dass ein und dasselbe Testmodell mit seinen Testfällen in unterschiedlichen Parameterkonfigurationen ohne weiteres verwendbar ist.



**Bild 8.3 Integration von Applikationskonfigurationen und -parametern**

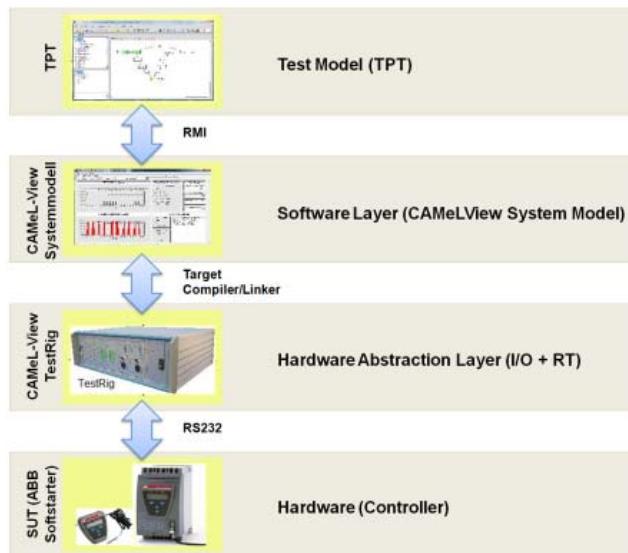
Hierzu wurden die TPT-Testmodelle prototypisch um Konzepte erweitert, die die Handhabung von Applikationskonfigurationen und -parametern ermöglichen. Hierzu zählen:

- Import/Export von Parameterdeklarationen
- Verwaltung von Applikationskonfigurationen
- Rollenmanagement für Parameter
- Unterstützung statischer Konfigurationen und dynamischer Konfigurationen
- Unterstützung primitiver und komplexer Parametertypen
- Testfall individuelles Überladen von Parameterdefinitionen

Mit Hilfe der erarbeiteten konzeptionellen Erweiterungen konnte exemplarisch nachgewiesen werden, dass sich TPT-Testmodelle parametrieren und damit flexibel an sich wandelnde Randbedingungen der Konfiguration und Applikation eines Steuerungs- oder Regelungssystem anpassen lassen, ohne dass hierfür neue, redundante Testmodelle oder neue Testfälle vonnöten sind. Dies erhöht die Effektivität der Testmodelle und die Wartbarkeit.

### 8.1.3 Integration TPT mit CameLView

Für TPT wurde eine Schnittstelle zu dem Mechatronic-Modellierungssystem CAMeL-View von iXtronics geschaffen, mit deren Hilfe sich an das CAMeL-View-TestRig angeschlossene Software oder Hardware über ein CAMeL-View-Systemmodell testen lässt.



**Bild 8.4 Architektureschichten**

Die gewählte Architektur sieht eine klare Layer-Trennung zwischen Hardware (Controller/ECU), Hardware-Abstraction-Layer (CAMEL-View TestRig), Software-Layer (CAMEL-View Systemmodell) und Testmodell (TPT) vor. Diese Trennung ermöglicht klar strukturierte und einfach wartbare Schnittstellen. Das Testmodell beschränkt sich auf die Definition der relevanten Tests auf Basis der logischen Signale. Diese Signale bilden die Schnittstelle für die Stimulation und Observation des zu testenden Systems. Sie sind im darunter liegenden Systemmodell in CAMEL-View modelliert. Das CAMEL-View-Modell bildet die logischen Systemsignale auf konkrete Hardware-Signale und -kanäle ab. Das Systemmodell wird anschließend auf der Target-Hardware kompiliert und ausgeführt. Über entsprechende I/O-Schnittstellen werden die Hardware-Signale mit dem eigentlichen zu testenden Controller ausgetauscht.

Das gewählte Konzept wurde in der ABB-Fallstudie evaluiert. Es konnte nachgewiesen werden, dass der modellbasierte Test von Industriesteuerungen mit dem beschriebenen Ansatz vergleichsweise einfach und praktikabel umsetzbar ist.

#### 8.1.4 Fallstudie Daimler

PikeTec hat gemeinsam mit anderen Partnern die Daimler-Fallstudie bei der Spezifikation und Revision der Anforderungen unterstützt sowie die Fallstudie beim Einsatz der erarbeiteten Konzepte und Tools begleitet. Der Schwerpunkt der Arbeiten von PikeTec im Rahmen der Daimler-Fallstudie lag in der Untersuchung der Effektivität von Testwiederverwendung bei der Durchgängigkeit von modellbasierten Tests von Model-in-the-Loop, Software-in-the-Loop bis Hardware-in-the-Loop auf der Basis des Testwerkzeugs TPT. Darüber hinaus wurde die Parameterunterstützung in Testmodellen evaluiert. Ein weiterer Schwerpunkt der Arbeiten im Kontext der Daimler-Fallstudie lag in der Untersuchung von Möglichkeiten der Prozess- und Datenintegration zwischen Testmanagement-Tools und modellbasiertem Test am Beispiel der Kopplung zwischen DanTe und TPT. (DanTe ist eine spezifische Ausprägung des weit verbreiteten Testmanagement-Werkzeugs Quality-Center von HP.) Dabei wurde untersucht, wie sich Daten- und Prozesselemente aus dem Testmanagement auf Bausteine modellbasierter Tests sinnvoll abbilden lassen.

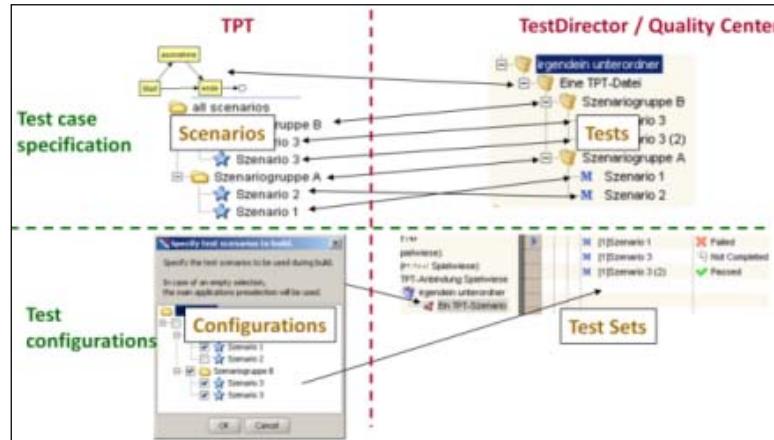


Bild 8.5 Anforderungsspezifikation mit TPT

### 8.1.5 Fallstudie ABB

Bei der ABB-Fallstudie hat PikeTec gemeinsam mit anderen Partnern bei der Spezifikation und Revision der Anforderungen mitgearbeitet sowie die Fallstudie beim Einsatz der erarbeiteten Konzepte und Tools begleitet. Der Schwerpunkt der Arbeiten von PikeTec im Rahmen der ABB-Fallstudie lag in der modellbasierten Testinfrastruktur für einen Softstarter. Hierzu wurde von iXtronics ein CAMEL-View- Modell zur physikalischen Kopplung des Softstarters mit dem TestRig-System erstellt. Das CAMEL-View-Modell erlaubt es mittels TTL-Signalen die Tastendrücke des externen Softstarter Keypads zu emulieren. Die Antworten des Softstarters auf diese simulierten Tastendrücke werden via RS232 Schnittstelle zum TestRig kommuniziert.

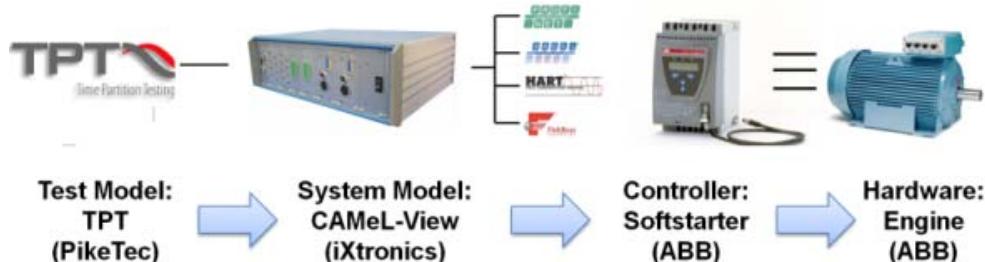


Bild 8.6 Testmodellierung für den Softstarter

In der Fallstudie wurden mehrere Tests exemplarisch evaluiert. Hierzu gehörte z.B. der Test der Umschaltung des Sprachmenüs des Softstarters: Durch simulierte Tastendrücke wurden die verschiedenen Menüstrukturen durchlaufen, die eine Umschaltung des Displaymenüs von deutscher Sprache auf z.B. auf spanische Menüs und wieder zurück zur deutschen Menüsprache vollautomatisch testen konnten. Die Testabläufe wurden hierbei in TPT modelliert. Das CAMEL-View System-Modell hat die Stimulationen in I/O-Signale für die Hardware übersetzt, die dann über RS232-Nachrichten an den Softstarter-Controller gesendet wurden. Außerdem wurden die Konfiguration und die Funktion der Rampen für das Anlaufen und Stoppen eines Motors mit dem Softstarter modellbasiert getestet.

## 8.2 VORAUSSICHTLICHER NUTZEN

PikeTec konnte im Rahmen von D-MINT die technologische Kompetenz sowie das Wissen bzgl. Marktbedürfnissen und Randbedingungen für den modellbasierten Test von Steuerungs- und Regelungssystemen deutlich ausbauen. Auf der Basis der Experimente zur Integration von TPT mit Hardware-in-the-Loop Systemen sind seitens PikeTec in 2010 und 2011 weiterführende Arbeiten geplant. Unter anderem ist eine weitergehende Kooperation mit iXtronics geplant. Ziel dieser Kooperation soll die

 <b>D-MINT</b> Deployment of Model-Based Technologies to Industrial Testing	<b>D-MINT Abschlussbericht</b>	Seite : 31 of 81 Version: 1.0 Datum: 30.06.2010 <hr/> Status : Final
---	--------------------------------	---

gemeinsame Verwertung der Ergebnisse der Werkzeugkopplung zwischen CAMEL-View und TPT sein, um die Kopplung der Werkzeuge als Produkt weiterzuentwickeln und für den Automobil- und Industrieautomatisierungsmarkt anbieten zu können. Außerdem plant PikeTec gegenwärtig gemeinsam mit der MBtech GmbH (Hersteller von PROVEtech:TA), die für die Daimler-Fallstudie erstellte prototypische Integration von TPT mit PROVEtech:TA in den kommenden zwei Jahren zu einer Produktlösung auszubauen. Darüber hinaus ist der Ausbau der Funktionalität von TPT in Bezug auf die Parameter- und Konfigurationsunterstützung in der Produktversion bis 2011 geplant.

### 8.3 FORTSCHRITT BEI ANDEREN STELLEN

Aufgrund der Veröffentlichungen und Präsentationen der Ergebnisse von D-MINT durch PikeTec konnte das Netzwerk mit potentiellen Partnern ausgebaut werden. Daraus ergaben sich beispielsweise auch erste Ideen und Gespräche zur Wiederverwendung der Konzepte in anderen Industriezweigen wie der Medizintechnik und der Luft- und Raumfahrtindustrie.

### 8.4 ERFOLGTE ODER GEPLANTE VERÖFFENTLICHUNGEN

Die Projektergebnisse sind, soweit nicht vertraulich, unter <http://www.d-mint.org> verfügbar. Weitere Veröffentlichungen sind:

- [58] Jens Lüdemann, Eckard Bringmann, Andreas Krämer, Messeausstellung, Embedded World 2008, Nürnberg, Germany, 2008
- [59] Eckard Bringmann, Automated model-based testing of control software with TPT, Embedded World Conference 2008, Nürnberg, Germany 2008
- [60] Eckard Bringmann, Andreas Krämer, Model-Based Testing of Automotive Systems, First International Conference on Software Testing, Verification, and Validation, ICST 2008, Lillehammer, Norway, 2008
- [61] Stefan Lachmann, Andreas Krämer, Systematischer Test des kontinuierlichen Verhaltens automobiler Softwaresysteme, 3. Tagung: Simulation und Test für die Automobilelektronik, Berlin, Germany 2008
- [62] Jens Lüdemann, Eckard Bringmann, Andreas Krämer, Automotive Testing Expo 2008, Stuttgart, Germany 2008
- [63] Krämer, Gut, Baderschneider, Systematischer Test von Fahrerassistenz-Systemen in einer virtuellen Verkehrsumgebung, Hardware-in-the-Loop Simulation, Kassel, Germany 2008
- [64] Jens Lüdemann, Eckard Bringmann, Andreas Krämer, Messeausstellung, Embedded World 2009 Nürnberg, Germany 2009
- [65] S. Weiß, Model based Control Testing using TPT, Webex Meeting D-Mint Webinar, Berlin, Germany 2009
- [66] Jens Lüdemann, Automatic ASAM MCD-3 supported model based testing, Technology Forum at the Automotive Testing Expo 2009, Stuttgart, Germany 2009
- [67] Jens Lüdemann, Eckard Bringmann, Andreas Krämer, Automotive Testing Expo 2009, Stuttgart, Germany 2009
- [68] Dr. Jens Lüdemann, Systematischer Softwaretest sicherheitskritischer Systeme Technology Forum DESIGN&ELEKTRONIK-Entwicklerforum »Embedded goes medical«, Leipzig, Germany 2009
- [69] Dr. Stefan Lachmann, Embedded Software Systeme systematisch und durchgängig Testen, DESIGN&ELEKTRONIK-Entwicklerforum »Embedded-System-Entwicklung« Ludwigsburg, Germany 2009
- [70] Dr. Eckard Bringmann, Testen von eingebetteten Steuerungs- und Regelungssystemen ASQF-Fachgruppen-Treffen Test, Berlin, Germany 2009

 Deployment of Model-Based Technologies to Industrial Testing	<b>D-MINT Abschlussbericht</b>	Seite : 32 of 81 Version: 1.0 Datum: 30.06.2010 Status : Final
---	--------------------------------	---

## 9. DAIMLER AG, FÖRDERKENNZEICHEN 01 IS 07 001 D

Im Rahmen von D-MINT war Daimler vor allem an der Entwicklung und dem Einsatz neuer Technologien zur Ermittlung von Testfällen aus Entwicklungsmodellen wie Architektur-, UML- und Simulink-Modellen interessiert. Diese neuen Technologien sollten zu einem Mehrwert gegenüber bereits in der Daimler-Forschung untersuchten Ansätzen zur modellbasierten Testfallermittlung führen. Mit Blick auf aus den Ansätzen resultierenden Testfällen könnte man auch sagen, dass Daimler auf der Suche nach neuartigen Testfällen war, d.h. solchen, die nicht mit den bereits untersuchten Ansätzen erzeugt werden. Diese Testfälle sollten dabei automatisch erzeugt werden, dies ist eine Anforderung, die üblicherweise mit dem modellbasierten Test verbunden wird.

Entsprechend wurden mehrere neuartige Verfahren in D-MINT untersucht und in mehreren Fallstudien aus dem Automotive-Bereich erprobt. In Zusammenarbeit mit anderen D-MINT-Partnern wurden zusätzlich Demonstratoren entwickelt, die auf verschiedenen europäischen Ausstellungen gezeigt wurden.

Daimler war in D-MINT auch als Leiter des Arbeitspakets Fallstudien aktiv, sowohl national als auch international (auf europäischer Ebene).

### 9.1 ERREICHTE ERGEBNISSE

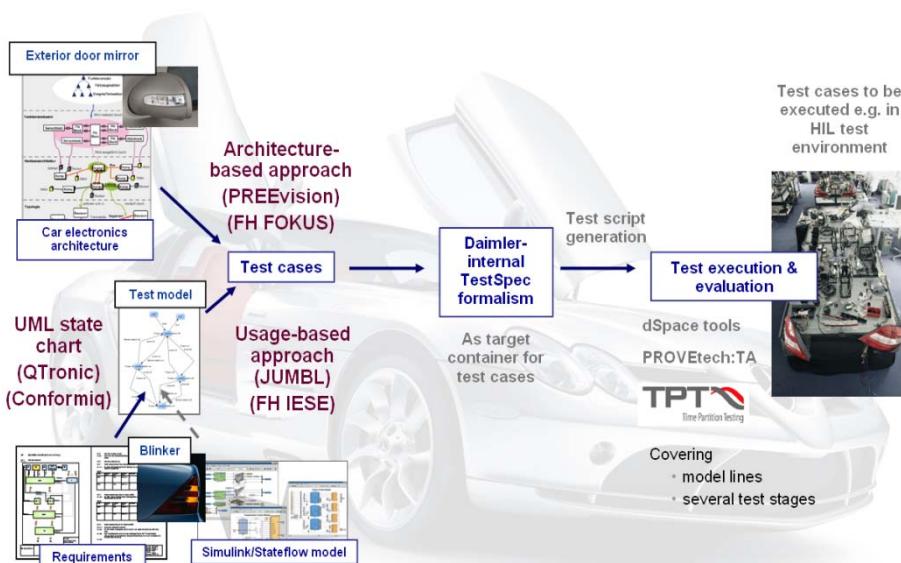
Wie erläutert war Daimler neben der inhaltlichen Arbeit in D-MINT auch Arbeitspaketsleiter für alle D-MINT-Fallstudien. Die inhaltlichen Ergebnisse sind in Abschnitt 7.1.1 und die Ergebnisse der Arbeitspaketleitung in Abschnitt 7.1.2 dargestellt.

#### 9.1.1 Inhaltliche Ergebnisse

Daimler hat in D-MINT die drei folgenden Ansätze zur modellbasierten Testfallermittlung untersucht (vgl. Bild 9.1):

1. **Architektur-basierte Testfallermittlung** auf der Basis von E/E-Architekturen, die bei Daimler zur Entwicklung Software-basierter Fahrzeugelektronik eingesetzt werden. Diese Arbeit wurde in enger Zusammenarbeit mit D-MINT-Partner Fraunhofer FOKUS durchgeführt.
2. **Benutzungs-basierte Testfallermittlung** auf Basis von speziell für den Test erstellten Modellen, die die möglichen Benutzungen des Testobjekts beschreiben. Diese Arbeit wurde in enger Zusammenarbeit mit D-MINT-Partner Fraunhofer IESE durchgeführt.
3. **Testfallermittlung auf der Basis von Zustandsmodellen**. Auch hier ist die Grundlage ein speziell für den Test entwickeltes Modell, das mögliche Zustände des Testobjekts beschreibt. Diese Arbeit wurde in enger Zusammenarbeit mit D-MINT-Partner Conformiq durchgeführt.

Die beiden letztgenannten Ansätze benutzen speziell für den Test erstellte Modelle, die die Sicht von Testern wiedergeben sollen. Sie sind Entwicklungsmodellen ähnlich, doch enthalten sie darüber hinaus speziell für den Test relevante Informationen. Diese Modelle werden auch als Testmodelle bezeichnet, um sie von anderen Modellen wie Entwicklungs- und Implementierungsmodellen abzugrenzen. Die Verwendung dieser Testmodelle ist ein momentan sehr innovatives Thema im Bereich Testen.



**Bild 9.1** Automotive case study

Zur Erprobung des ersten Ansatzes wurde die Steuerung eines externen Spiegels verwendet, zur Erprobung der anderen beiden Ansätze die Steuerung eines Blinkers.

Der Reifegrad der Ansätze war zu Beginn des Projektes wie folgt:

- Die Arbeit an der **Architektur-basierten Testfallermittlung** startete während der Laufzeit des Projekts. D.h. die Basis für diese Technik wurde während der Projektlaufzeit entwickelt. Aufsetzen konnten wir allerdings auf einige theoretische Arbeiten auf dem Gebiet.
- Die beiden anderen beschriebenen Technologien waren bereits vorhanden, Tools zur Unterstützung dieser Ansätze existierten auch bereits. Allerdings befanden sich die Tools für die **Benutzungs-basierte Testfallermittlung** noch im Prototyp-Stadium, während das Tool für die **Zustands-basierte Testfallermittlung** bereits kommerziell von Conformiq vertrieben wurde.

Entsprechend dem für D-MINT geplanten Projektverlauf wurden zur Entwicklung und Erprobung dieser drei Technologien Fallbeispiele inklusive Anforderungen an von anderen D-MINT-Partnern bereitzustellende Test-Technologien wie beschrieben identifiziert und dokumentiert. Nach Bereitstellung der Technologien wurden diese in den Fallbeispielen eingesetzt, anschließend wurde dieser Einsatz ausgewertet. Unterstützt wurde diese Auswertung von Messdaten, die während der Erprobung gesammelt und mit dem Zustand vor dem Einsatz der D-MINT-Technologien verglichen wurden. Basis für diese Messungen waren Metriken, die vor der Erprobung in den Fallbeispielen festgelegt wurden. Darüber hinaus wurden Demonstratoren erstellt, die die Anwendung der D-MINT-Technologien in den Fallbeispielen veranschaulichen sollten. Bild 9.2 zeigt einen von mehreren Demonstratoren für die Fahrzeug-Fallstudien.



**Bild 9.2 Automotive Demonstrator (ITEA-Symposium Madrid 2009)**

Die in D-MINT erreichten inhaltlichen Ergebnisse auf Basis der Evaluation der drei Ansätze sind die folgenden:

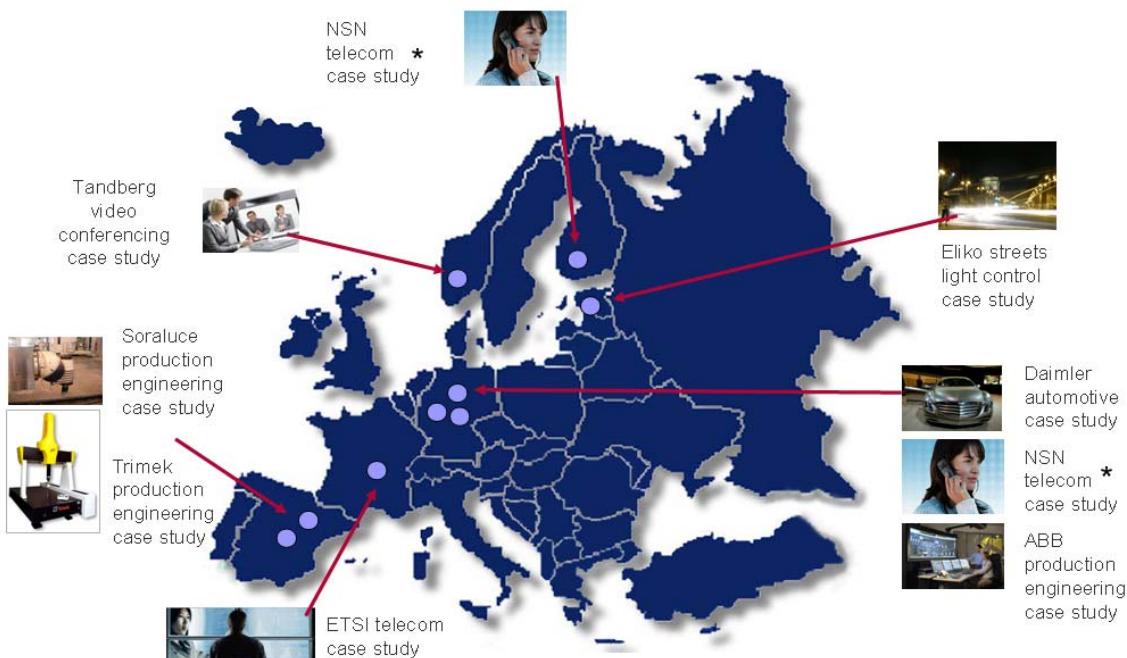
1. **Architektur-basierte Testfallermittlung:** dieser Ansatz ist sehr vielversprechend, da er prinzipiell in der Lage ist, neuartige Testfälle zu erzeugen, die darauf beruhen, dass in der Architektur nicht offensichtliche Beziehungen zwischen Komponenten sichtbar werden. Diese Information kann u.a. genutzt werden, um zu prüfen, ob sich indirekt miteinander in Beziehung stehende Komponenten stören könnten. Der Ansatz ist wie gesagt sehr vielversprechend, und es wurden auch im D-MINT-Projekt sehr gute theoretische Erfahrungen damit gemacht, doch benötigt der Ansatz noch weitere Entwicklungsarbeit, die über die Laufzeit des D-MINT-Projekts hinausgehen. Darüber hinaus ist für eine Anwendung in der Industrie eine solide Tool-Unterstützung notwendig, die zusätzlich geschaffen werden muß.
2. **Benutzungs-basierte Testfallermittlung:** mit diesem Ansatz wurden sehr gute Erfahrungen gemacht. Es handelt sich hierbei um einen sehr soliden Ansatz, der im Grunde leicht verständlich ist. Es wird zunächst ein Benutzungsmodell erstellt, das anschließend um für den Test relevante Informationen erweitert wird (z.B. Benutzungshäufigkeiten und Kritikalitäten oder Kosten von Benutzungen). Das resultierende Testmodell lässt sich grafisch darstellen und als Basis für Diskussionen mit Entwicklern und Testern nutzen. Ein Schwachpunkt bei diesem Ansatz ist seine Tool-Unterstützung, die sich noch im Prototyp-Stadium befindet und deshalb in dieser Form in der industriellen Praxis noch nicht einsetzbar ist. Mit entsprechender Tool-Unterstützung jedoch wäre ein Einsatz in der Fahrzeug-Entwicklung bei Daimler denkbar.
3. **Testfallermittlung auf der Basis von Zustandsmodellen:** Hierfür wurde eine Tool-gestützte Technologie von D-MINT-Partner Conformiq untersucht. Es stellte sich heraus, dass die von diesem Tool unterstützte Modellierung, die aus dem Telekom-Bereich stammt, die Anforderungen aus dem Fahrzeuggbereich nicht optimal erfüllt, da sich die Modellierung eher für diskrete als für kontinuierliche Systeme eignet. Fahrzeug-Systeme sind jedoch vorwiegend kontinuierlich. Darüber hinaus wurde zur selben Zeit ein ähnliches Tool einer anderen D-MINT-externen Firma bei Daimler untersucht, das

geeigneter war, die Anforderungen von Daimler zu erfüllen. Die gemeinsame Entwicklung am Ansatz mit Conformiq wurde deshalb in D-MINT nicht weiter verfolgt.

Zusammengefasst lässt sich sagen, dass Daimler zwei Ansätze untersucht hat, die tatsächlich zu neuen Testfällen führen können. Dies jedenfalls haben die Fallbeispiele in D-MINT gezeigt. Da jedoch bei beiden der entsprechende professionelle Tool-Support (noch) fehlt, konnte diese Annahme nicht in einer breiten Erprobung innerhalb der Daimler-Entwicklung bestätigt werden. Trotzdem werden diese beiden Ansätze weiter von der Daimler-Forschung untersucht und bei entsprechendem Reifegrad später vermutlich auch in der Daimler-Entwicklung zum Einsatz kommen. Voraussetzung hierfür ist allerdings, dass sich die Technologien nachweislich auch für reale, d.h. komplexe Fahrzeug-Systeme einsetzen lassen und eine geeignete Tool-Unterstützung bereitgestellt wird.

### 9.1.2 D-MINT Arbeitspaketleitung

Daimler leitete sowohl national als auch international das D-MINT-Arbeitspaket 1 (Fallstudien und Demonstratoren). Insgesamt wurden in D-MINT acht Fallstudien in sechs Ländern durchgeführt (in den Anwendungsfeldern Automotive, Telecom, Production Engineering, Street Lights Control, Video Conference Systems).



**Bild 9.3: D-MINT-Fallstudien**  
(\* die Telekom-Fallstudie war auf zwei Länder verteilt)

Wie schon bei der Daimler-Fallstudie beschrieben, bestand die Durchführung der Fallstudien aus den Schritten

- Festlegung und Dokumentation der Fallstudie inkl. Spezifikation von Anforderungen an D-MINT-Technologien,
- Demonstrator-Entwicklung,
- Fallstudien-Auswertungen (2 Iterationen).

Sechs dieser Fallstudien wurden als offizielle und zwei als inoffizielle Fallstudien im Projekt behandelt. Die inoffiziellen Fallstudien kamen von Partnern, die keine Förderung erhielten (aus den Ländern Norwegen und

Frankreich). An die beiden letztgenannten Fallstudien wurden deshalb nicht dieselben hohen Anforderungen wie an die offiziellen Fallstudien gestellt.

Die Arbeit mit den Fallstudien war insofern sehr erfolgreich, da

- alle Fallstudien wie geplant durchgeführt werden konnten,
- für alle sechs offiziellen Fallstudien Demonstratoren aufgebaut und verteilt in allen drei ITEA-Symposien, die während der D-MINT-Laufzeit stattfanden, gezeigt wurden (siehe Bild 9.4),
- D-MINT mit diesen Demonstratoren den „ITEA 2009 Symposium Exhibition Award 2009“ gewonnen hat,
- mit Hilfe der Fallstudien gezeigt werden konnte, dass der Einsatz der D-MINT-Technologien zu Kostenreduktion (beim Test), einer höheren Überdeckung des jeweiligen Testobjekts und zu einer verbesserten Komplexitätsbewältigung (beim Test) führen kann.





**Bild 9.4: D-MINT-Ausstellungen auf ITEA-Symposien**

Da Daimler sowohl national als auch international das D-MINT-Arbeitspaket 1 (Fallstudien und Demonstratoren) leitete, kamen im Projekt neben der Arbeit an der eigenen Fallstudie eine Reihe weiterer Aufgaben hinzu. Hierzu gehörten u.a. die Unterstützung aller Fallstudienpartner bei ihren Arbeiten, die Vorbereitung und Leitung von WP1 Sessions auf diversen D-MINT-Meetings, die ständige Pflege und Kontrolle des WP1 Deliverable Plans und die Erstellung von WP1-Berichten als Input für die periodischen ITEA- und BMBF-Berichte. Die Planung und der Stand der Arbeiten in diesem Arbeitspaket wurden von Daimler auf verschiedenen Meetings und über andere Kommunikationskanäle (Telefon, Emails usw.) regelmäßig kommuniziert

## 9.2 VORAUSSICHTLICHER NUTZEN

Wie bereits erläutern, wird die Daimler-Forschung zwei der drei untersuchten Ansätze weiter verfolgen: die Architektur- und die Benutzungs-basierte Testfallermittlung, letzteres auf der Basis von Testmodellen. Noch ist nachzuweisen, dass diese beiden Ansätze auch für umfangreichere und damit reale Fahrzeug-Systeme anwendbar sind. Bei erfolgreichem Nachweis wäre dann noch die Frage nach einer industriell anwendbaren Tool-Unterstützung zu klären. Fraunhofer IESE plant bereits, sein Tool zur Unterstützung der Benutzungs-basierten Testfallermittlung in Richtung industriellen Einsatz weiterzuentwickeln. Die Lösung von Fraunhofer FOKUS zur Unterstützung der Architektur-basierten Testfallermittlung basiert momentan auf einem Prototyp, der ein kommerzielles Architektur-Design-Tool erweitert. Bis aus diesem Prototypen ein industriell anwendbares Tool wird, müssen noch einige Entwicklungsarbeiten durchgeführt werden. Günstig wäre es jedoch, es fände sich ein Tool-Hersteller, der Interesse daran hätte, die beschriebenen Tools zur Industrie-Reife zu entwickeln.

Gegeben der Fall, diese Tools ständen bereit, wäre es sehr wahrscheinlich, dass Daimler aus deren Einsatz einen Mehrwert schöpfen könnte. Doch ist auch so schon ein Mehrwert entstanden, da die Methoden durch die Daimler-Forschung bereits in den Daimler-Entwicklungsgebieten propagiert werden.

## 9.3 FORTSCHRITT BEI ANDEREN STELLEN

Bei Daimler sind während der Laufzeit von D-MINT weitere Ansätze zum modellbasierten Test untersucht worden, so u.a. auf der Basis von Zustandsmodellen und Interaktionsdiagrammen. Diese Ansätze sind disjunkt zu den in D-MINT untersuchten Ansätzen und führen zusammen mit den D-MINT-Ansätzen zu einem erweiterten Technologie-Pool bei Daimler.

 Deployment of Model-Based Technologies to Industrial Testing	D-MINT Abschlussbericht	Seite : 38 of 81 Version: 1.0 Datum: 30.06.2010 Status : Final
--	-------------------------	---

## 9.4 ERFOLGTE ODER GEPLANTE VERÖFFENTLICHUNGEN

Die Projektergebnisse sind, soweit nicht vertraulich, unter <http://www.d-mint.org> verfügbar.

Es ist darüber hinaus geplant, eine weitere Veröffentlichung als schriftlichen Beitrag für ein Journal zu erstellen, die das D-MINT-Projekt als Ganzes und seine Ergebnisse beschreibt.

Weiterhin ist geplant, einen weiteren D-MINT-Demonstrator auf dem kommenden ITEA-Symposium in Gent, 2010, zu zeigen.

## 10. FRAUNHOFER IESE, FÖRDERKENNZEICHEN 01 IS 07 001 E

### 10.1 ERREICHTE ERGEBNISSE

Das Fraunhofer IESE ist Forschungspartner im Projekt. Der Fokus der Arbeiten lag auf dem Arbeitspaket 2, wo die modellbasierte Testtechniken entwickelt wurden. Das IESE leitete auch noch die Disseminationsaktivitäten des Projekts im Arbeitspaket 5,

Im Rahmen des AP 2 und AP 5 wurden die folgenden wesentlichen Ergebnisse erarbeitet:

- Risikobasiertes statistisches Testen
- Verbindung von statistischen und kombinatorischen Testansätzen
- Messkonzept für die Bewertung der Fallstudien
- Konzept für Coaching und Training

#### 10.1.1 Thematische Ergebnisse

Die thematischen Ergebnisse werden im Folgenden dargestellt.

##### 10.1.1.1 Risikobasiertes statistisches Testen

Das modellbasierte statistische Testen wurde bisher auf Basis so genannter Benutzungsprofile durchgeführt. Diese Profile bilden die tatsächliche Benutzung des Testobjekt durch seine Umgebung ab und ermöglichen die automatisierte Generierung von repräsentativen Testfällen. Die Verwendung von Risikoprofilen und der Aufbau von Modellen für den risikobasierten Test stellen eine Innovation dar, die im D-Mint Projekt entwickelt wurde.

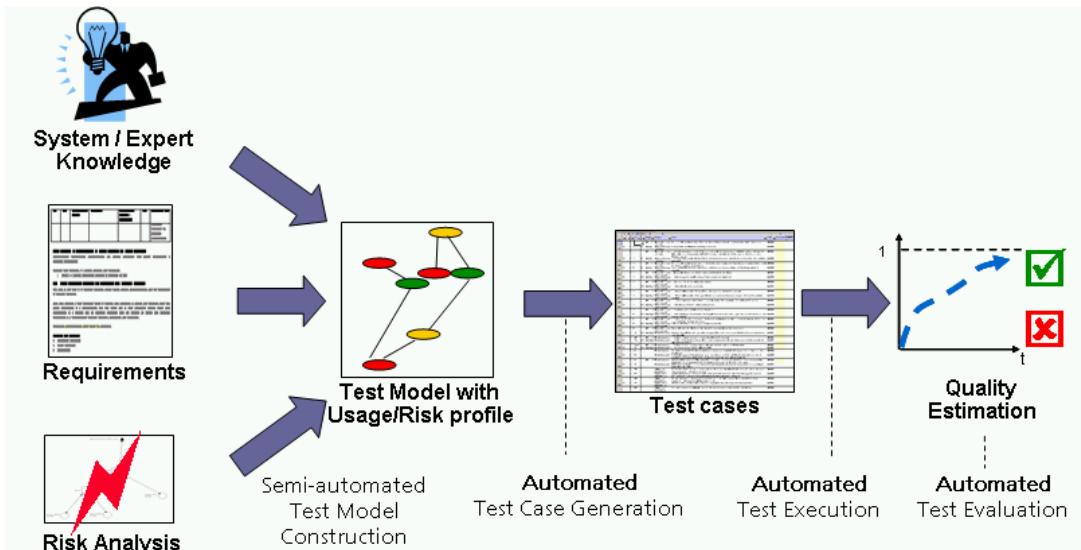
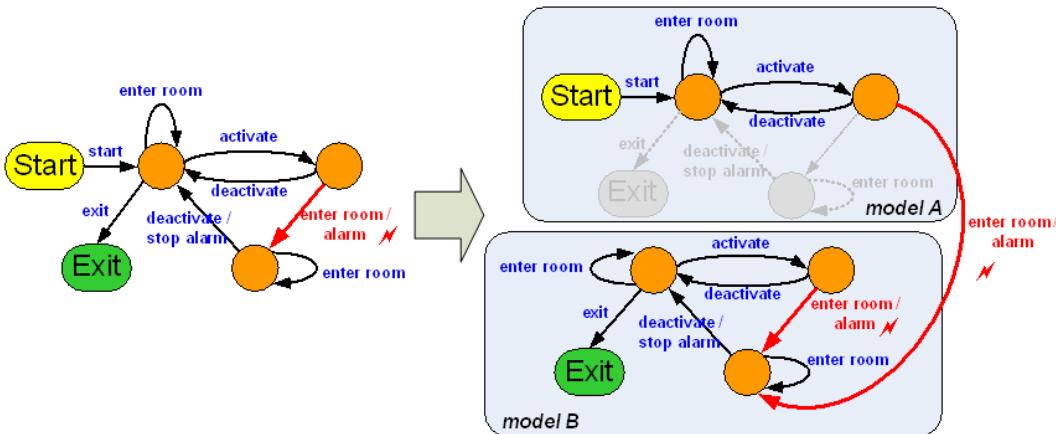


Bild 10.1: Risikobasiertes statistisches Testen

Das Bild 10.1 zeigt die Phasen des risikobasierten Testansatzes. Aus den Produktanforderungen wird systematisch ein Testmodell erstellt, das die testrelevanten funktionalen Systemanforderungen als Eingabe-Ausgabe-Sequenzen formal beschreibt. Die Elemente des Testmodells sind mit Gewichten annotiert um deren Wichtigkeit für den Test abzubilden. Für das risikobasierte Testen werden zusätzlich die Ergebnisse der Risikoanalyse verwendet um kritische Systemfunktionen und Komponenten zu identifizieren. Diese Informationen werden verwendet um im Testmodell genau auf die Aspekte zu fokussieren. Das Testmodell wird so aufgebaut, dass bei der Testfallableitung die kritische Aspekte besonders berücksichtigt und abgedeckt werden.



**Bild 10.2: Risikobasiertes Testmodell**

Das Bild 10.2 zeigt den Aufbau eines risikobasiertes Testmodells. In der Risikoanalyse werden die kritische Systemfunktionen und Komponenten ermittelt. Daraus können dann die relevanten Systemeingaben und – ausgaben identifiziert werden.. Im Testmodell werden die Elemente, die kritische Ein- oder Ausgaben beschreiben markiert (wie die Transition *enter room / alarm* im Beispiel der Abbildung). Im nächsten Schritt (rechts in der Abbildung) wird das Modell so erweitert, dass bei der Testfallableitung möglichst viele Testfälle mit den kritischen Elementen generiert werden. In dem Beispiel traversiert jeder Testfall die vorher als kritisch eingestufte Transition *enter room / alarm*.

Für die Testfallgenerierung werden sowohl strukturorientierte Verfahren zur systematischen Modellüberdeckung als auch stochastische Verfahren für die Berücksichtigung des Risikoprofils verwendet.

Test Case	Cost, Length	Probability	Risk Coverage
1	22	1.227E-15	630
2	19	1.325E-13	650
3	10	6.340E-07	320
4	41	2.808E-32	1120
5	9	5.120E-07	250

Test Case	Cost, Length	Probability	Risk Coverage
4	41	2.808E-32	1120
2	19	1.325E-13	650
1	22	1.227E-15	630
3	10	6.340E-07	320
5	9	5.120E-07	250

Test Case	Cost, Length	Probability	Risk Coverage
4	41	2.808E-32	1120
2	19	1.325E-13	650
1	22	1.227E-15	630
5	9	5.120E-07	250
3	10	6.340E-07	320

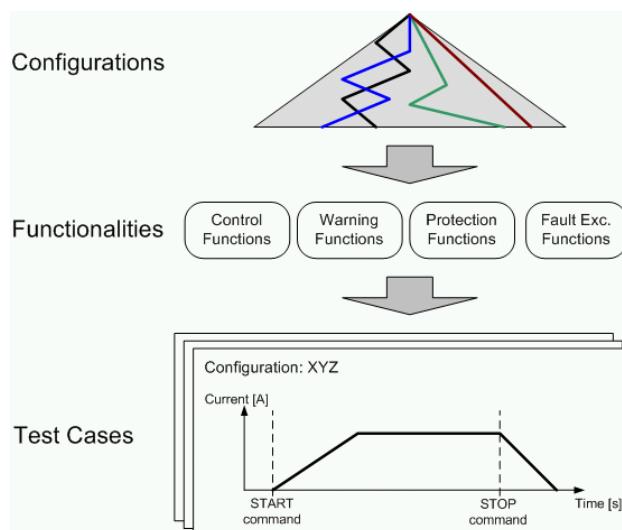
**Bild 10.3: Risikobasierte Priorisierung von Testfällen**

Das Bild 10.3 zeigt die Bewertung der abgeleiteten Testfälle und deren nachfolge risikobasierte Priorisierung. Jeder Testfall besteht aus einer Sequenz von Testschritten, in denen das Testobjekt mit Eingabedaten stimuliert wird. Die Ausführungskosten jedes Testfalls werden berücksichtigt, falls entsprechende Kosteninformationen verfügbar sind. Im einfachsten Fall werden die gleichen Kosten für jeden Testschritt angenommen. Die Kosten für die Ausführung eines Testfalls hängen dadurch direkt von dessen Länge ab. Jeder Testschritt wird zusätzlich bezüglich seiner Risikoabdeckung bewertet. Die Risikoabdeckung des Testfall wird aus den Bewertungen seiner Testschritte berechnet. Die Testfälle mit der höchsten Gesamtbewertung sollen gemäß der risikogetriebenen Teststrategie als erstes ausgeführt. Die Testfälle können nun nach ihrer Risikoabdeckung sortiert werden. Eventuelle Kostenvorgaben des Projektmanagements werden bei der Testfallauswahl berücksichtigt. Im Beispiel der Abbildung wird dadurch der Testfall 3 aus der Testsuite entfernt.

Für die Validation des Ansatzes wurden projektbegleitend prototypische Werkzeuge entwickelt, die wesentliche Punkte umsetzen, insbesondere die systematische Konstruktion von Testmodellen.

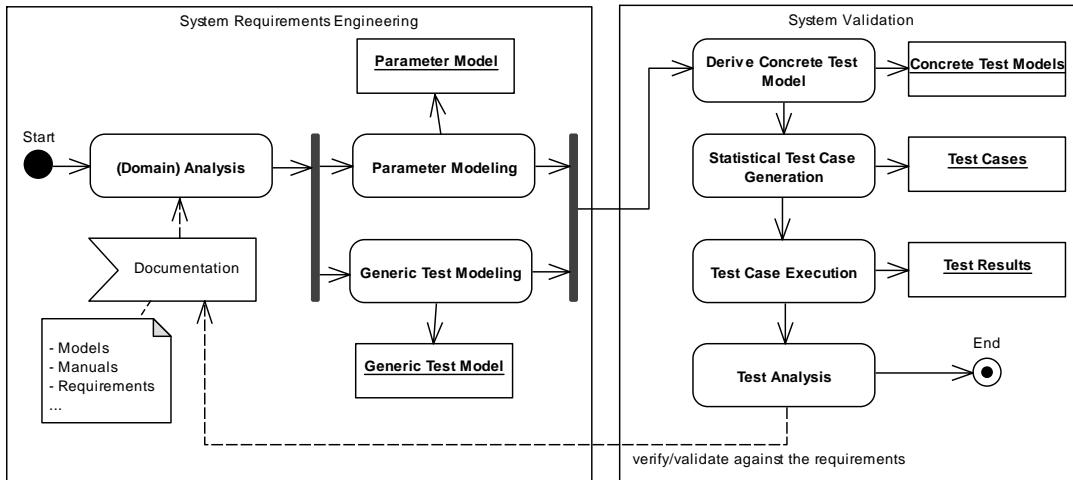
### **10.1.1.2 Verbindung von statistischen und kombinatorischen Testansätzen**

Beim modellbasierten Testen werden insbesondere Verhaltens- und Umgebungsmodelle für die automatisierte Testfallableitung verwendet. Für den Test von Systemen mit vielen unterschiedlichen Konfigurationen werden Testansätze benötigt, die systematisch die Produktkonfigurationen und –funktionen abdecken. Im Rahmen des D-Mint Projekts wurden dafür statistische Testansätze mit kombinatorischen Testansätzen kombiniert. Die kombinatorischen Testansätze überdecken effizient die unterschiedlichen Produktkonfigurationen. Die statistischen Testansätze dienen der Überprüfung der Systemfunktionen unter Berücksichtigung der Benutzungshäufigkeit und des Risikos. Konfigurationsabhängige Funktionen werden ebenfalls abgebildet.



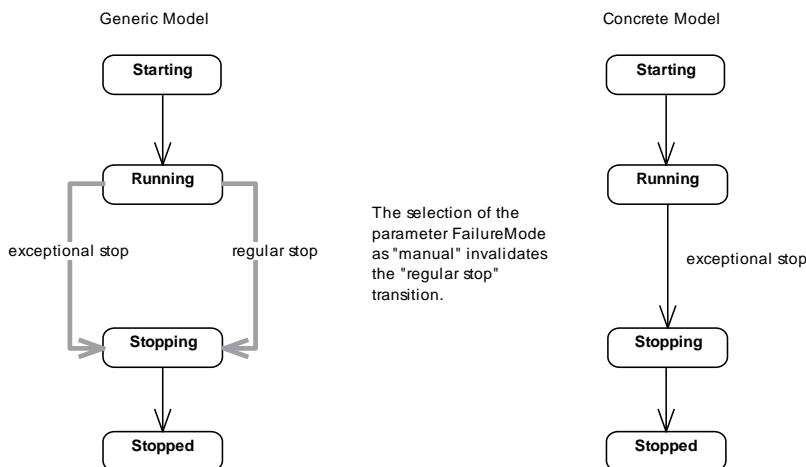
**Bild 10.4: Konzept zur Verbindung von statistischen und kombinatorischen Testansätzen**

Das Bild 10.4 zeigt das Konzept zur Verbindung von statistischen und kombinatorischen Testansätzen anhand der ABB-Fallstudie im Projekt. Im Rahmen der Fallstudie wurde der Testansatz auf Systemebene angewendet. Er umfasst dadurch die Phasen Systems Requirements Engineering und System Validation. Im ersten Schritt wurden die Konfigurationsparameter und deren gegenseitige Abhängigkeiten bestimmt. Diese Informationen bilden das Konfigurationsmodell. Die konkreten Konfigurationen für die Testdurchführung werden mit kombinatorischen Verfahren aus dem Konfigurationsmodell abgeleitet. Im nächsten Schritt wurden aus den Produktanforderungen und der Produktdokumentation die relevanten Systemfunktionen mit den entsprechenden Eingabe-Ausgabe-Sequenzen bestimmt. Für die ABB-Fallstudie waren das insbesondere die Steuerungs- und Sicherheitsfunktionen des Testobjekts. Im letzten Schritt werden dann für jede generierte Konfiguration gültige Testfälle aus dem funktionalen Testmodell abgeleitet.



**Bild 10.5: Schritte des kombinierten Testansatzes**

Das Bild 10.5 zeigt die einzelnen Schritte des Testansatzes. Nach der Analyse von Dokumentation und Produktanforderungen werden die Konfigurationen und die generischen funktionalen Anforderungen in den Schritten *Parameter Modeling* und *Generic Test Modeling* modelliert. Aus dem generischen Testmodell werden konfigurationspezifische Testmodelle abgeleitet. Testfälle werden automatisiert aus den konkreten Testmodellen abgeleitet, ausgeführt und ausgewertet. Aufgedeckte Fehlverhalten können zu Änderungen im Testobjekt (Programmcode) oder dessen Anforderungen oder Dokumentation führen.



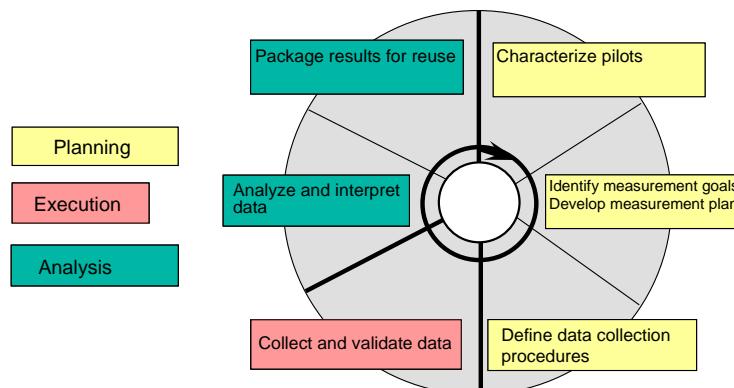
**Bild 10.6: Generische und konfigurationsspezifische Modelle**

Das Bild 10.6 zeigt die Beziehung zwischen generischen und konfigurationsspezifischen Modellen. Im generischen Modell (links) sind alle konfigurationsspezifischen Einstellungen und Funktionen berücksichtigt. Im Beispiel gibt es zwei Alternativen für den Zustandsübergang vom *Running* nach *Stopping*. In Abhängigkeit von der Konfiguration wird ein normaler (*regular*) Stop oder gesondert behandelter Stop ausgeführt. Im konkreten Testmodell (rechts), das auch dem generischen Modell abgeleitet wurde, sind alle Alternativen und Mehrdeutigkeiten aufgelöst. Für jede Konfiguration gibt es damit eine entsprechende Umsetzung der funktionalen Anforderungen, hier ist es der gesondert behandelte Stop (*exceptional stop*).

In der Fallstudie konnte gezeigt werden, dass der entwickelte Ansatz die Konfiguration und Produktfunktionen effizient abdeckt und testet.

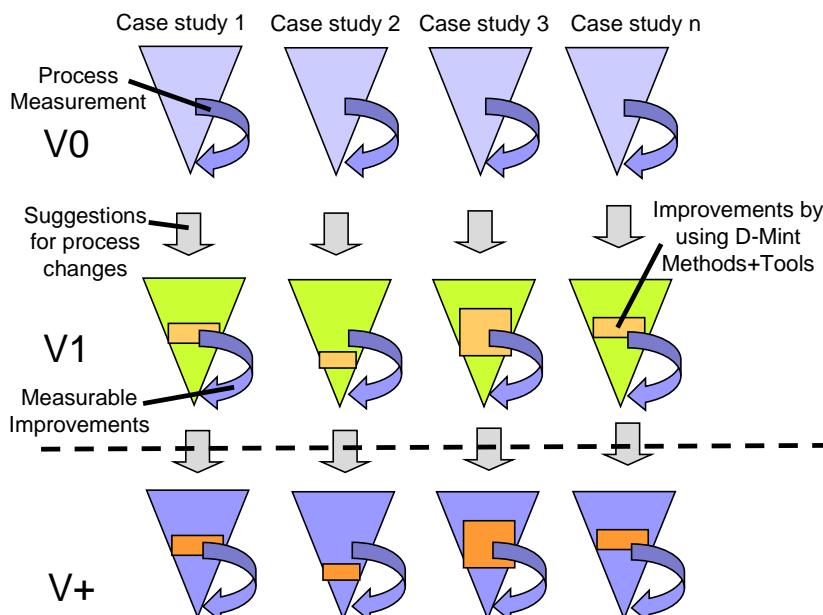
### 10.1.1.3 Messkonzept für die Bewertung der Fallstudien

Ein Ziel des Projekts war der empirische Vergleich der neu entwickelten oder weiter entwickelten modellbasierten Testtechnologien. Dafür hat das Fraunhofer IESE zusammen mit den Industriepartnern für jede Fallstudie einen speziellen Messplan entwickelt. Die Messpläne ermöglichen die existierenden, etablierten Testansätze mit den neuen modellbasierten Testansätzen zu vergleichen. Sie enthalten eine Reihe von Maßzahlen, die insbesondere die Kosten, den Aufwand und den Nutzen (bspw. Fehleraufdeckungsrate) der Testtechnologien ausdrücken.



**Bild 10.7: Phasen der Messung gemäß GQM-Ansatz**

Für die Ermittlung der Maßzahlen wurde der Goal-question-metric (GQM)-Ansatz gewählt. Er beschreibt ein systematisches Vorgehen zur Definition geeigneter Metriken bei der Messung von Prozessen. Die einzelnen Schritte sind im Bild 10.7 gezeigt. Die Messpläne werden aufgestellt, die Daten werden erhoben und ausgewertet. Auf Basis der Ergebnisse der Datenanalyse kann dann eine eventuell notwendige weitere Iteration der Messung vorbereitet werden.



**Bild 10.8: Messung und Bewertung der Fallstudien**

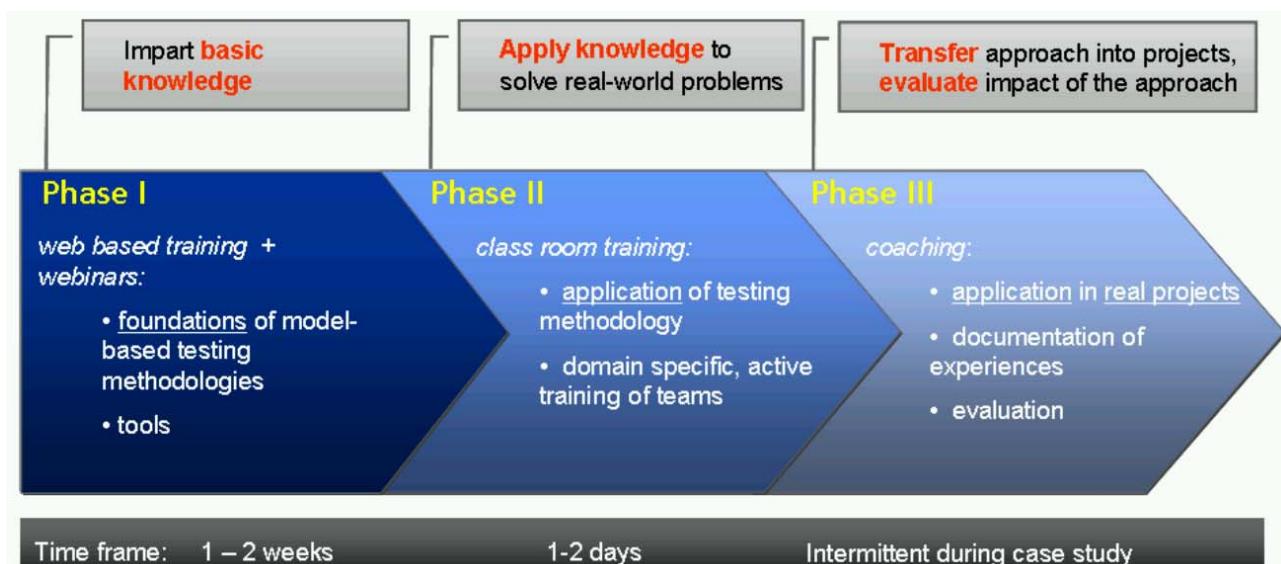
Das Bild 10.8 zeigt den Ansatz für die empirische Messung und Auswertung der einzelnen Fallstudien. Zu Beginn wurden die Messpläne aufgestellt. Im nächsten Schritt wurden die Maßzahlen für die Bewertung der existierenden, etablierten Testansätze in den Fallstudien erhoben. Auf Basis der Messung wurde die Verbesserungspotentiale identifiziert und durch den Einsatz von modellbasierten Testtechniken umgesetzt.

Für die abgeänderten Testprozesse mit den neuen Technologien wurden dann wieder Maßzahlen gemäß des Messplans erhoben. Die Maßzahlen aus den beiden Iterationen wurden dann verglichen. Außerdem wurden mögliche Verbesserungen in den Testprozessen für eine weitere Iteration vorgeschlagen. Diese Iteration ist jedoch nicht mehr Teil des Projekts.

Die Ergebnisse der Messungen wurden insbesondere für die interne und externe Verbreitung und Verwertung der Projektergebnisse verwendet (Arbeitspakete 4 und 5).

#### **10.1.1.4 Konzept für Coaching und Training**

Die Coaching- und Trainingsaktivitäten in D-Mint basieren auf einem 3 Phasen-Ansatz. Dieser Ansatz ermöglicht einen schrittweisen Technologietransfer, beginnend mit Grundlagenschulungen von Technologien und Werkzeugen. Im zweiten Schritt werden domänenspezifische Probleme von den Technologieanwendern gelöst. Der abschließende Schritt beinhaltet den Technologietransfer in reale Industrieprojekte. Die Technologieanwender werden dabei von Werkzeug- und Technologieexperten unterstützt und gecoacht.



**Bild 10.9: Phasen des Konzepts für Coaching und Training**

Das Bild 10.9 zeigt die 3 Phasen des Konzepts. In jeder Phase wurden unterschiedliche Schulungsunterlagen in Abhängigkeit von der Domäne und dem Wissen der Schulungsteilnehmer erstellt. Diese Aktivitäten umfassen u.a. face2face Seminare, web-basierte Schulungen mit State-of-the-art Technologien, Webinare (Online-Seminare) und Universitätskurse. Das Highlight des D-Mint Trainingskonzepts war die periodische Webinarreihe zum Thema „Modellbasiertes Testen“ mit Beiträgen von verschiedenen Partnern und Ländern. Das Fraunhofer IESE hat die Umsetzung der Webinar-Reihe koordiniert.

#### **10.1.1.5 Verbreitung der Ergebnisse**

Das Fraunhofer IESE hat die Projektergebnisse aus D-Mint auf internationalen Konferenzen und Workshops veröffentlicht. Eine Liste der relevanten Veröffentlichungen ist unten zu finden. Das Fraunhofer IESE war Mitbegründer des MoTiP-Workshops<sup>1</sup> im Jahr 2008. Dessen Thema ist „Model-based Testing in Practice“. Der Schwerpunkt liegt auf der Diskussion zum Thema MBT zwischen Forschung und Industrie um die Verbreitung zu unterstützen. Im Juni 2010 findet die 3. Auflage des Workshops in Paris an der ECMFA statt. Weitere MoTiP-Workshops für die folgenden Jahre sind geplant. Daneben wurden zusammen mit dem Fraunhofer FOKUS verschiedene Tutorien zu MBT und zum D-Mint Projekt gehalten.

<sup>1</sup> Link: [http://www.fokus.fraunhofer.de/en/motion/news\\_events/veranstaltungsreihen/motip/index.html](http://www.fokus.fraunhofer.de/en/motion/news_events/veranstaltungsreihen/motip/index.html)

 <b>D-MINT</b> <small>Deployment of Model-Based Technologies to Industrial Testing</small>	<b>D-MINT Abschlussbericht</b>	Seite : 45 of 81 Version: 1.0 Datum: 30.06.2010 Status : Final
---	--------------------------------	---

Das Projekt wurden auch mehreren Messen präsentiert, u.a. den ITEA Symposia 2008, 2009 und 2010, wo das Fraunhofer IESE jeweils einen Demonstrator zum modellbasierten statistischen Testen ausstellte. Auf dem letzten ITEA Symposium 2010 in Madrid D-Mint für die Projektpräsentation auf dem Messestand den ITEA Exhibition Award erhalten.

### 10.1.2 Organisatorische Arbeiten

Auf Ebene des europäischen Projekts hat das Fraunhofer IESE das Arbeitspaket 5 (Dissemination) geleitet. Zudem wurden halbjährlich Berichte zum Stand und Ausblick des Arbeitspakets für ITEA erstellt.

## 10.2 VORAUSSICHTLICHER NUTZEN

Die im Vorhaben entwickelten modellbasierten Techniken passen sich gut in das vom IESE angebotene Dienstleistungsspektrum ein und werden in die Services zum modellbasierten Testen mit aufgenommen. Sie liefern dem IESE im Bereich des statistischen und risikobasierten Testens einen nennenswerten Kompetenzgewinn. Im Rahmen von Schulungen und Beratungsdienstleistungen werden die Ergebnisse weiter verbreitet und in die Praxis transferiert.

Zurzeit werden mögliche Kooperationen und Projekte in der Medizin- und Eisenbahntechnik diskutiert, Das zeigt die breite Anwendbarkeit der entwickelten Technologien. Weiterhin bietet die Kooperation innerhalb des Projektes die Möglichkeit, weitere Partner für die spätere Zusammenarbeit sowohl in industriellen Projekten als auch für Forschungskooperationen zu gewinnen.

Daneben wird am Fraunhofer IESE ein Technologielabor aufgebaut, in dem u.a. Softwaresysteme im Auftrag des Kunden getestet werden können. Dabei spielen die entwickelten modellbasierten Testtechnologien eine große Rolle.

## 10.3 FORTSCHRITT BEI ANDEREN STELLEN

Auf internationaler Ebene gab es eine umfassende Kooperation mit den Partnern in Spanien, Frankreich und Estland. Mit einigen internationalen Partnern wird das IESE in anderen Forschungsprojekten die Zusammenarbeit ausbauen.

Durch die Fallstudien aus verschiedenen Domänen konnte die breite Anwendbarkeit der modellbasierten Testtechniken gezeigt werden. Bei Messeauftritten wurden die D-Mint-Ergebnisse mit eindrucksvollen Demonstratoren präsentiert. Der Höhepunkt war das ITEA Symposium 2010 in Madrid, wo das Projekt den ITEA Exhibition Award erhielt.

Während und nach Projektlaufzeit wurden die entwickelten Technologien und Ergebnisse aus D-Mint in Industrieprojekten angewendet und weiter in die Praxis transferiert. Basierend auf den Ergebnissen und Erfahrungen wurden neue Themen im Bereich systematisches Testen und Qualitätssicherung identifiziert, die im Rahmen von weiteren Forschungsprogrammen bearbeitet werden sollen.

## 10.4 ERFOLGTE ODER GEPLANTE VERÖFFENTLICHUNGEN

Publikation des Fraunhofer IESE im Rahmen von D-MINT:

- [71] R. Eschbach, Sequence Based Specification and Statistical Testing, Siemens TS Software Initiative Workshop, Braunschweig, Germany, May 24, 2007
- [72] C. Robinson-Mallett, Model-based Testing, Siemens TS Software Initiative, Braunschweig , Germany, May 24, 2007

- [73] C. Robinson-Mallett, R.-M. Hierons, J. H. Poore, P. Liggesmeyer, Using Communication Coverage Criteria and Partial Model Generation to assist Software Integration Testing , Software Quality Journal, Vol.16, 2008
- [74] T. Bauer, T. Beletski, F. Böhr, R. Eschbach, D. Landmann, J. H. Poore, From Requirements to automated statistical testing, 4th International ICSE workshop on Software Engineering for Automotive Systems, SEAS 2007, Minneapolis, USA, May 26, 2007
- [75] T. Bauer, Model-based statistical testing of technical software systems, Systematic Testing Conference, Düsseldorf, Germany, October 22, 2007
- [76] C. Robinson-Mallett, R.-M. Hierons, J. H. Poore, T. Bauer, Using partial models to support the testing of distributed systems, The 11th IASTED International Conference on, Cambridge-MA, USA, November 19-21, 2007
- [77] D. Streitferdt, P. Nenninger, C. Bilich, F. Kantz, T. Bauer, R. Eschbach, Model-based Testing in the Automation Domain, Safety Enabled, 1st Workshop on Model-based Testing in Practice - MoTiP 2008, Berlin, Germany, 12-Jun-08, Fraunhofer IRB,
- [78] T. Bauer, F. Böhr, R. Eschbach, On MiL, HiL, Statistical Testing, Reuse, and Efforts, 1st Workshop on Model-based Testing in Practice - MoTiP 2008, Berlin, Germany, 12-Jun-08, Fraunhofer IRB
- [79] Robert Eschbach, Risikobasiertes statistisches Testen sicherheitskritischer eingebetteter Systeme, 7th SafeTRANS Industrial Day, Friedrichshafen Germany, 19.11.09
- [80] F. Zimmermann, R. Eschbach, J. Kloos, T. Bauer. Risikobasiertes statistisches Testen (German article), Workshop paper, 28. TAV-Treffen, Dortmund, Germany, 2009
- [81] F. Zimmermann, R. Eschbach, J. Kloos, T. Bauer, Risk-based Statistical Testing: A refinement-based approach to the reliability analysis of safety-critical systems, 2th European Workshop on Dependable Computing Toulouse, France, 14-15 May 2009
- [82] J. Kloos, R. Eschbach, Generating System Models for a Highly Configurable Train Control System Using A Domain-Specific Language: A Case Study, Workshop paper, 5th Workshop on Advances in Model Based Testing (A-MOST 2009) at the 2nd International Conference on Software Testing, Verification, and Validation (ICST'09), Denver, Colorado, USA, 2009
- [83] T. Bauer, R. Eschbach, M. Größl, T. Hussain, D. Streitferdt and F. Kantz, Combining Combinatorial and Model-based Test Approaches for Highly Configurable Safety-critical Systems, 2nd Workshop on Model-based Testing in Practice (MoTiP 2009), Enschede, The Netherlands, 23 June 2009, Link

#### D-MINT Tutorien und Workshops auf internationalen Konferenzen

- [84] T. Bauer, H. Eichler, A. Rennoch, 1st Workshop on model-based testing in practice (MoTiP'08) at the 4th European Conference on Model-Driven Architecture Foundations and Applications (ECMDA 2008), Berlin, Germany, 12 June 2008, MoTiP'08 Workshop Proceedings, Publisher: Fraunhofer IRB Verlag, ISBN 978-3-8167-7624-6, 12 June 2008
- [85] T. Bauer, A.I Rennoch, Ina Schieferdecker., Model-based Testing and its use in the ITEA2 research project D-MINT, 21st International Conference on Software & Systems Engineering and their Applications 2008, ICCSEA 2008, 9 Dec 2008
- [86] T. Bauer, A. Rennoch, TTCN-3 and its role and usage in MBT from the D-MINT, TTCN-3 user conference, Sophia Antipolis, France, 3. June 2009
- [87] T. Bauer, H. Eichler, A. Rennoch, S. Wieczorek, 2nd Workshop on model-based testing in practice (MoTiP'09) at the 5th European Conference on Model-Driven Architecture Foundations and Applications (ECMDA 2009), Enschede, The Netherlands, 23 June 2009; MoTiP'09 Workshop Proceedings, CTIT Workshop Proceedings Series WP09-07, ISSN 0929-0671, Enschede, The Netherlands, 23 June 2009
- [88] T. Bauer, H. Eichler, M.-F. Wendland, S. Wieczorek, 3rd Workshop on model-based testing in practice (MoTiP'10) at the 6th European Conference on Model-Driven Architecture Foundations and Applications (ECMFA 2010), Paris, France, 16 June 2010

## 11. TESTING TECHNOLOGIES, FÖRDERKENNZEICHEN 01 IS 07 001 F

Testing Technologies hat sich auf die Entwicklung und Vermarktung von Testwerkzeugen und Testlösungen spezialisiert. Das technologische Kernkonzept ist TTCN-3 (Testing and Test Control Notation), eine von ETSI (European Telecommunication Standards Institute) standardisierte Testspezifikations- und Testimplementierungssprache. Die Designprinzipien von TTCN-3 erlauben deren Anwendung auf viele verschiedene Technologien und deren Einsatz in den unterschiedlichsten Domänen. Mit der TTworkbench bietet Testing Technologies eine hoch innovative Testplattform für TTCN-3 an, die den gesamten Zyklus der Testentwicklung und -ausführung unterstützt. Im Rahmen von D-MINT wurde die Weiterentwicklung von TTworkbench im Bereich des modellbasierten Testens durch die Integration von UTP (UML Testing Profile) fortgesetzt. Dabei geht es vornehmlich um die Transformation von Testmodellen in UTP nach TTCN-3 und deren automatische Ausführung in TTworkbench.

Testing Technologies' Engagement in D-MINT konzentrierte sich auf die Erweiterung der modellbasierten Testtechnologie UTP, sowie auf die Unterstützung und Durchführung der Fallstudien von ABB und IDEKO/Solaruce gemeinsam mit anderen Partnern in D-MINT.

### 11.1 ERREICHTE ERGEBNISSE

#### 11.1.1 Modellbasiertes Testen mit UTP

Während modellbasierte Softwareentwicklung fest in den heutigen Softwareentwicklungsprozessen verankert ist, haben sich modellbasierte Konzepte im Bereich des Testens nach wie vor nicht in ausreichendem Maße durchgesetzt. Mit dem UML Test Profile (UTP) wurde jedoch eine Entsprechung auch im Bereich des Testens gefunden. UTP definiert darüber hinaus eine Ableitung nach TTCN-3, um so eine automatisierte Testausführung von UTP basierten Testfällen zu realisieren.

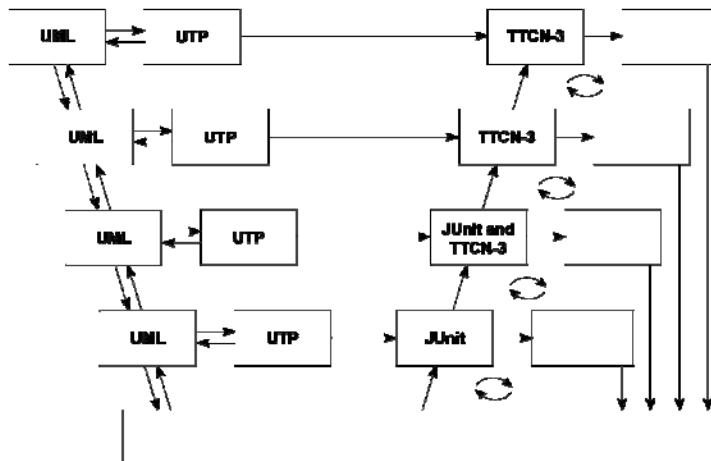
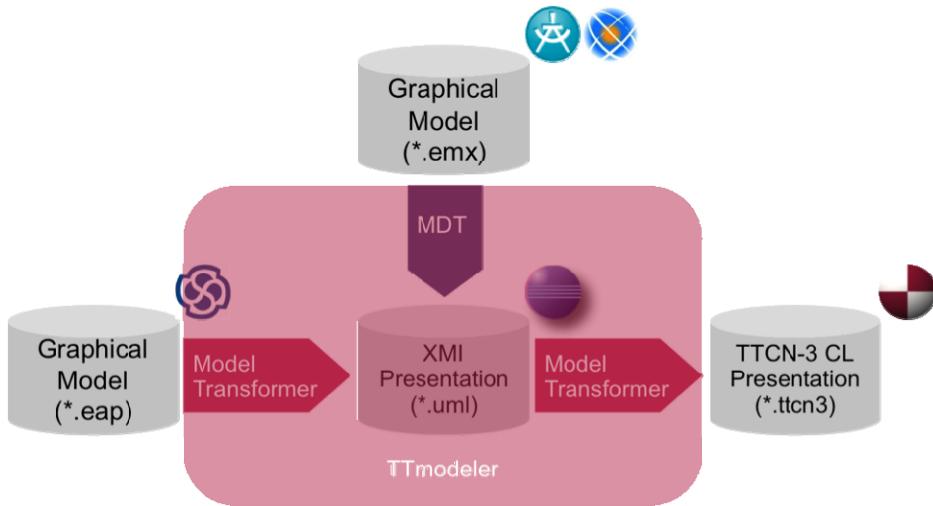


Bild 11.1 Modellbasieter Entwicklungsprozess

Testing Technologies hat im Rahmen von D-MINT eine UTP Umsetzung mit den UML Werkzeugen IBM Rational Software Architect und im europäischen UML Framework Projekt Topcased realisiert. Die in UTP modellierten Testfälle wurden über TTCN-3 zur Ausführung gebracht. Dazu wurde via Modelltransformation die TTCN-3 Testentwicklungs- und -ausführungsplattform TTworkbench angebunden.



**Bild 11.2 Modelltransformation UTP → TTCN-3**

### 11.1.2 UTP Umsetzung und Erweiterung

Durch die Mitarbeit und Unterstützung in den Fallstudien wurden neue Erkenntnisse sowohl an die Umsetzung der UTP Konzepte, als auch hinsichtlich deren Erweiterung und Ergänzung gewonnen und umgesetzt. Exemplarisch dafür stehen die State Machine Diagramme, die ein höheres über die Sequence Diagramme hinausgehendes Abstraktionsniveau bieten und somit dem Anspruch modellbasierter Konzepte eher entsprechen. Die folgenden UTP Features und deren Ableitung nach TTCN-3 wurden im Rahmen von D-MINT umgesetzt:

- Sequence Diagramme
- State Machine Diagramme
- Static Configuration
- Parameterized Messages, Calls/Replies
- UML Alternativen
- Timer Operationen (start/read/stop/timeout/running)
- Exception Behandlung

### 11.1.3 Fallstudie IDEKO/Soraluce

In der Fallstudie von IDEKO/Soraluce geht es um das Testen von Industrieautomatisierungssystemen, im speziellen CNC gesteuerte Fräsmaschinen. Im Fokus stand in dieser Fallstudie die Verzahnung von modellbasierter Software- und Testentwicklung, sowie die automatisierte Testausführung in den verschiedenen Phasen des Entwicklungsprozesses.

Bei der IDEKO/Soraluce Fallstudie hat Testing Technologies an der Spezifikation der Anforderungen mitgewirkt, sowie beim Einsatz der Modellierungskonzepte und Werkzeuge begleitet. Testing Technologies unterstützte hier insbesondere in der Testdesignphase, aber auch in der Testimplementierung und Adaption an die zu testenden Systeme.

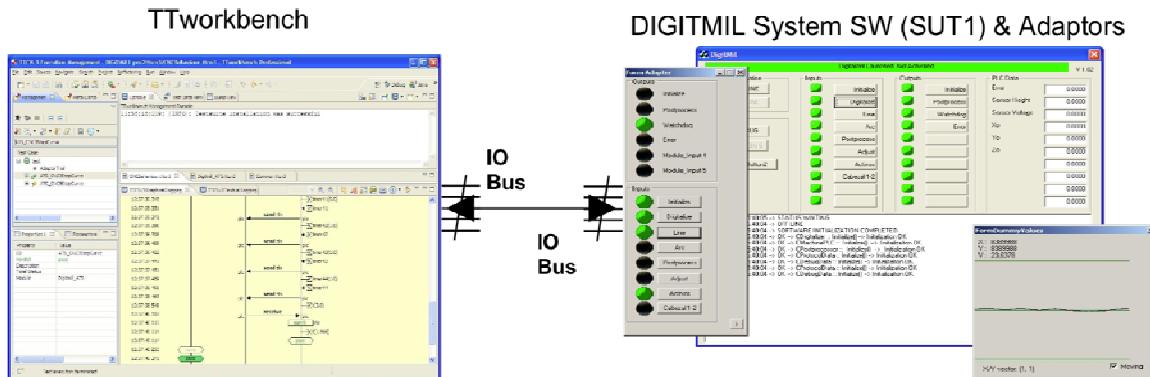


Bild 11.3 Softwarekomponententest

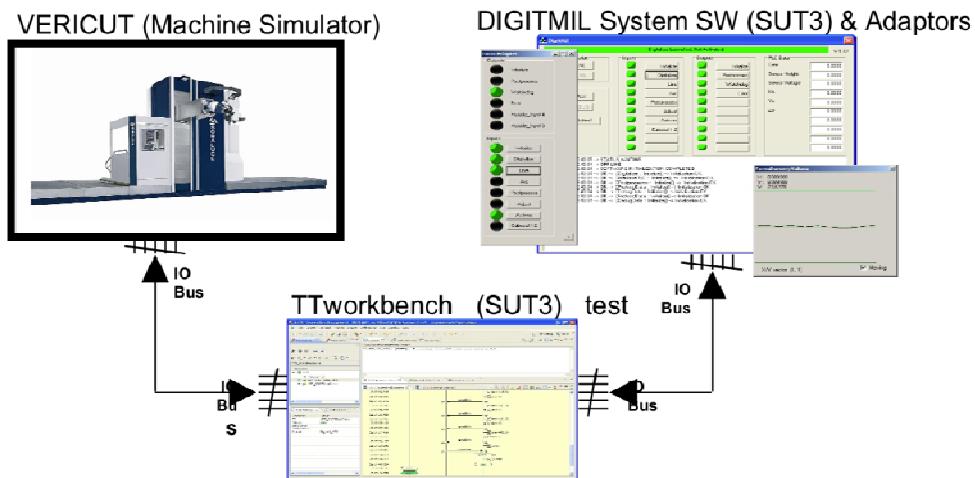


Bild 11.4 Integrationstest

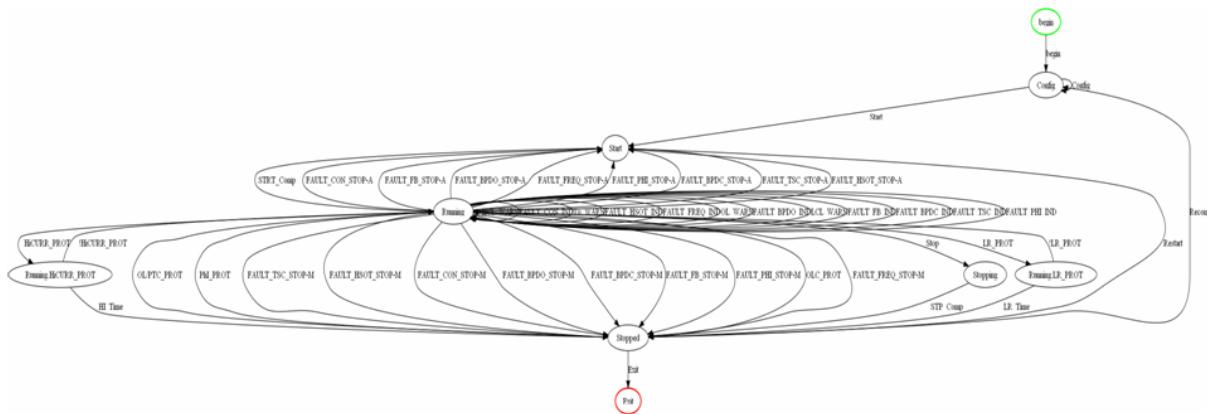
Insbesondere die Möglichkeit der Wiederverwendung der Modelle und der daraus abgeleiteten Tests für die verschiedenen Phasen ohne weitere Modifikation oder Anpassung derselben und die damit einhergehenden Kosteneinsparungen stellen das herausragende Ergebnis dieser Fallstudie dar.

#### 11.1.4 Fallstudie ABB

Die Fallstudie von ABB stammt ebenfalls aus dem Bereich der Industrieautomatisierung, hier steht das Testen eines Softstarters im Fokus. Die ABB Fallstudie zeichnet sich durch die Vielzahl der erprobten Konzepte und Partner aus. Testing Technologies hat gemeinsam mit anderen Partnern an der Anforderungsanalyse mitgewirkt und im Rahmen der Fallstudie einzusetzenden Konzepte und Werkzeuge beraten. Testing Technologies war in diesem Rahmen für den Ansatz UTP → TTCN-3 und dessen Umsetzung mit der Werkzeugkette TTmodeler → TTworkbench verantwortlich.

In diesem Rahmen wurde darüber hinaus eine Kopplung von TTworkbench und iXtronics CAMeL-View umgesetzt. Hierzu wurde von iXtronics ein CAMeL-View Modell zur physikalischen Kopplung des Softstarters mit der TestRig Testhardware erstellt. Das CAMeL-View Modell erlaubt es die Tastendrücke des externen Softstarter Keypads zu emulieren. Die Antworten des Softstarters auf diese simulierten Tastendrücke werden via RS232 Schnittstelle zum TestRig kommuniziert. TTworkbench kommuniziert mit CAMeL-View über eine RMI Schnittstelle.

Neben der Modellierung der Testfälle mittels UTP wurde im Rahmen der ABB Fallstudie noch ein weiterer Ansatz in Kooperation mit Fraunhofer IESE evaluiert. Dazu wurde eine Integration mit dem Fraunhofer IESE Werkzeug MBST zur Generierung von Testfällen aus statistischen Modellen umgesetzt. MBST generiert dabei aus statistischen Modellen TTCN-3 Code, der ebenfalls über die Kopplung iXtronics CAMEL-View – TTworkbench zur Ausführung gebracht wird.

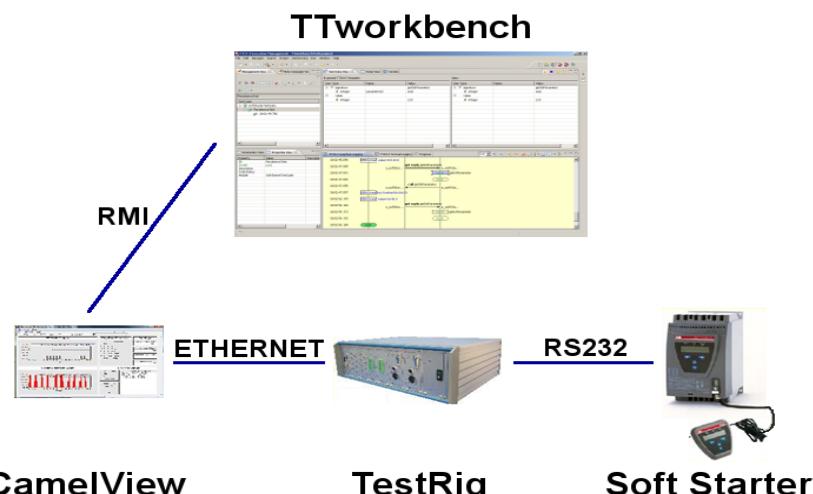


**Bild 11.5 Statistisches Modell**

### 11.1.5 Integration TTmodeler/TTworkbench mit CAMEL-View

Die Kopplung zwischen der TTCN-3 Testplattform TTworkbench und dem iXtronics Mechatronic-Modellierungssystem CAMEL-View wurde über eine RMI Schnittstelle realisiert, mit deren Hilfe sich an das CAMEL-View/TestRig angeschlossene Software oder Hardware über ein CAMEL-View Systemmodell testen lässt.

In TTworkbench wird das Testverhalten als Sequenz von Testbefehlen in TTCN-3 beschrieben. Diese Sequenz kann entweder aus UTP oder einem statistischen Modell generiert werden. Die Befehlsaufrufe werden von CAMEL-View interpretiert und an das Testmodell für das System unter Test (SUT) weitergeleitet. Dort werden sie in entsprechende Befehlsequenzen zur Ansteuerung des SUT umgesetzt. Die Befehlsequenzen werden in Echtzeit verarbeitet. Ein Befehlsaufruf aus der TTworkbench kann einem einzelnen Befehl im CAMEL-View Testmodell entsprechen oder aus einer Sequenz von Befehlen bestehen. Die Antwort des SUT wird auf gleichem Wege via Testmodell an die TTworkbench zurück gesendet und dort ausgewertet.



**Bild 11.6 Integration TTworkbench – CAMEL-View**

Dieses Konzept erlaubt es kontinuierliche Signale zu generieren und auf kontinuierliche Signale in Echtzeit zu reagieren und diese zu evaluieren. In der ABB-Fallstudie hat sich dieser Ansatz bewährt. Voraussichtlicher Nutzen

## **11.2 FORTSCHRITT BEI ANDEREN STELLEN**

Im Rahmen von D-MINT konnten aus den verschiedenen Fallstudien wertvolle Erkenntnisse gewonnen werden. Sowohl bezüglich des Funktionsumfangs einer MBT Werkzeugkette, als bezüglich Usability und möglicher Einsatzszenarien. Da Testing Technologies' eigene Wurzeln im Bereich der Telekommunikation liegen, war insbesondere die Beteiligung an Fallstudien aus anderen Domänen besonders wertvoll. Das UTP Werkzeug TTmodeler konnte mit den gewonnenen Erkenntnissen bis zur Produktreife weiterentwickelt werden und steht in Kürze für den Einsatz in modellbasierten Testprojekten zur Verfügung. Darüber hinaus ist geplant, die im Rahmen der ABB Fallstudie umgesetzte Werkzeugkopplung zwischen iXtronics CAMEL-View und TTworkbench weiter auszubauen und bei gemeinsamen Projekten einzusetzen.

## **11.3 ERFOLGTE ODER GEPLANTE VERÖFFENTLICHUNGEN**

Durch die Mitarbeit in D-MINT ist das Interesse an der Kopplung von TTCN-3 mit modellbasierten Techniken, insbesondere mit UTP, insgesamt angewachsen. Dies betrifft sowohl den TTCN-3 Heimatmarkt Telekommunikation, als auch andere Märkte, wie Medizintechnik, Automotive und Industrieautomatisierung. Erste Gespräche lassen hier auf ein steigendes Interesse an UTP/TTCN-3 schließen.

## **11.4 ERFOLGTE ODER GEPLANTE VERÖFFENTLICHUNGEN**

Die Projektergebnisse sind, soweit nicht vertraulich, unter <http://www.d-mint.org> verfügbar. Weitere Veröffentlichungen sind:

- [89] Theofanis Vassiliou-Gioles, Model-based testing with U2TP and TTCN-3, 4th European Conference on Model Driven Architecture Fondations and Applications, Berlin, Germany, 2008
- [90] Bogdan Stanca-Kaposta, Stephan Pietsch, Model-based testing with UTP and TTCN-3 and its Application to HL7, Conference on Quality Engineering in Software Technology, Potsdam, Germany, 2008
- [91] Bogdan Stanca-Kaposta, Executable Test-Models – From UML Test Profile to Test Report, ASQF Fachgruppe Software-Test Berlin-Brandenburg, Berlin, Germany, 2009
- [92] Theofanis Vassiliou-Gioles, Testautomatisierung mit TTCN-3 am Beispiel des UML Testprofils, Software & Systems Quality Conference (SQC), Düsseldorf, Germany, 2009
- [93] Alexander Lorenz, Derivation and Automatic Generation of TTCN-3 Test Cases from UML State Machine Test Models, Master Thesis, FernUniversität Hagen, Germany, 2008

## 12. ABB AG, FÖRDERKENNZEICHEN 01 IS 07 001 G

Im Rahmen von D-MINT hat ABB als Industriepartner aus der Automatisierungsdomäne in Zusammenarbeit mit den D-MINT Partnern die Einsetzbarkeit modellbasierter Testmethoden analysiert. Als Ergebnis dieser Analyse wurden die für die Automatisierungsdomäne passenden Methoden gewählt und in ein Gesamtvorgehen integriert. Dieses Vorgehen wurde anhand einer Fallstudie bewertet.

### 12.1 ERREICHTE ERGEBNISSE

Auf Basis der Analyse modellbasierter Testmethoden konnten die D-MINT Testmethoden auf die spezifischen Bedingungen der Automatisierungsdomäne zugeschnitten werden. Dieses Vorgehen wurde im Rahmen der ABB-Fallstudie angewandt und bewertet.

Die drei Hauptergebnisse, Entwicklung der Fallstudie, Entwicklung einer angepassten modellbasierten Testmethode und schließlich deren Bewertung werden in den folgenden drei Abschnitten vorgestellt. Diese Ergebnisse wurden in enger Zusammenarbeit mit Fraunhofer IESE, iXtronics, PikeTec und Testing Technologies erzielt, alle ebenfalls im Rahmen des ITEA Projektes D-MINT gefördert.

#### 12.1.1 Fallstudie

Die Fallstudie der Automatisierungsdomäne ist um einen ABB Sanftanlauf (auch Softstarter) herum aufgebaut. Ein Softstarter wird genutzt, um Elektromotoren sanft anlaufen zu lassen, d.h. die Drehzahl des Motors entsprechend vom Nutzer wählbarer Anlaufkurven von Null auf Maximaldrehzahl zu regeln. Die softwaregesteuerte Regelung basiert auf der Ansteuerung von Thyristoren (vergleichbar mit Schaltern), die zu beliebigen Zeitpunkten geschlossen werden können und damit die Sinuswelle von Strom/Spannung an den Motor weitergeben. Mit jedem Nulldurchgang der Sinuswelle öffnen Thyristoren automatisch und stellen dann wieder einen offenen Schalter dar. Damit wird der zum Elektromotor weitergegebene Anteil jeder Sinushalbwelle von dem Softstarter gesteuert. Über die vom Nutzer gewählte Anlaufkurve errechnet der Softstarter wie viel einer jeden Sinushalbwelle zu jedem Zeitpunkt der Anlaufkurve an den Motor weitergegeben wird, z.B. ein linearer Anstieg der Drehzahl über 10 Sekunden.

Es war zu klären wie ein strukturiertes modellbasiertes Testvorgehen aussehen muss, um die aktuell existierenden Testfälle sinnvoll zu ergänzen und gleichzeitig die hohe Qualität der Produkte stabil zu halten. Über die Anforderungsanalysephase wurden die vielversprechenden Ansätze der modellbasierten Testdomäne ausgewählt. Vier der Ansätze wurden in der Fallstudie angewandt und evaluiert.

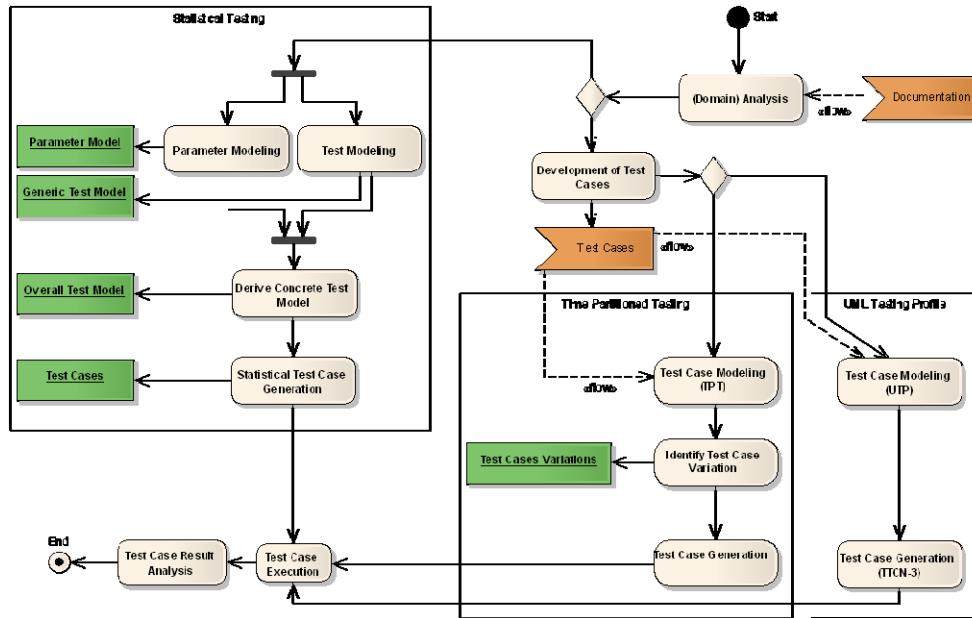
- Das statistische Testen nutzt Zustandsdiagramme erweitert um Wahrscheinlichkeiten. In Zusammenarbeit mit Fraunhofer IESE wurden zwei Wahrscheinlichkeitsprofile entwickelt, um den Anforderungen der Automatisierungsdomäne Rechnung zu tragen.
- Geräte in der Automatisierungsdomäne weisen eine Vielzahl von einstellbaren Parametern auf. Der Test dieser Geräte erfordert ein Parametermodell, welches in Zusammenarbeit mit Fraunhofer IESE unter Nutzung des „Pairwise Testing“ Ansatzes entwickelt wurde.
- „Time Partitioned Testing“ (TPT) nutzt Zustandsdiagramme mit Variantenunterstützung für unterschiedliche Testfallszenarien. Das entsprechende Werkzeug von PikeTec ist mit dem TestRig von iXtronics integriert, um die Testfälle auszuführen. TPT wurde in der Fallstudie in Zusammenarbeit mit PikeTec und iXtronics genutzt.
- Testfälle wurden mit Hilfe des „UML Testing Profile“ beschrieben, um daraus automatisch ausführbare Testfälle abzuleiten. Dabei werden Sequenzmodelle in TTCN-3 Code umgesetzt, welcher dann auf dem Testsystem ausführbar ist. Dieser Lösungsansatz wurde zusammen mit Testing Technologies und iXtronics entwickelt.



**Bild 12.1: Demonstrator**

Die Ausführung der generierten Testfälle benötigt eine Testumgebung mit Softstarter, einem Echtzeitrechner für die konkrete Testfallausführung und Überwachung der Ausführung mittels Sensoren, einem Elektromotor, einer regelbaren Last (ein zweiter Elektromotor betrieben als Bremse) und dem Modell einer Anwendung, im Falle des in Bild 12.1 gezeigten Demonstrators handelt es sich um ein Förderband, welches vom sanft anlaufenden Elektromotor bewegt wird (z.B. um ein Kippen der Teile auf dem Förderband zu verhindern).

## 12.1.2 Angepasste Modellbasierte Testmethode



**Bild 12.2: D-MINT Modellbasiertes Testen in der Automatisierungsdomäne**

Der modellbasierte D-MINT Testansatz für die Automatisierungsdomäne basiert, wie in Bild 12.2 dargestellt auf drei unterschiedlichen Wegen der Testfallgenerierung und stellt die domänenspezifische Ableitung des in D-MINT entwickelten architekturgetriebenen Testansatzes [104] dar.

Über den **statistischen Testansatz** mit einer Erweiterung um "Pairwise Testing" werden "Black-Box" Testfälle für das zu testende System generiert, unter Nutzung eines Testmodells (eines statistischen Zustandsautomaten) und eines Parametermodells aus dem die notwendigen Parametersätze für jeden Testfall generiert werden. Mit einer Kombination von Parametermodellen mit formal definierbaren einschränkenden Bedingungen zwischen Parametern, und dem "Pairwise Testing" Ansatz (→ Recursive Backtracking) konnten Reduktionen der möglichen Parameter Permutationen von über 99% erzielt werden [94]. Das Testmodell basiert auf Dokumenten wie z.B. Anforderungsdokumenten oder Benutzerhandbüchern. Diese Informationen werden manuell in Sequenzen umgewandelt, welche das Verhalten des Systems wiedergeben und in eine „Sequence Based Specification (SBS)“ münden. Mit Hilfe des Werkzeugs Pro-toSeq werden die Sequenzen automatisch in Zustandsdiagramme umgewandelt. Zusätzlich zum Zustandsdiagramm werden typische Anwendungsszenarien, die Risikoverteilung und die Sicherheitsanalyse / -bewertung der Systemkomponenten benötigt, um die Wahrscheinlichkeitsprofile der Transitionen des Zustandsautomaten zu entwickeln. Mit den Wahrscheinlichkeiten, dem Zustandsautomaten und dem Werkzeug JUMBL ist die Ableitung der Testfälle für jedes der Profile möglich. Die Testfälle und Parametersätze des "Pairwise Testing" Ansatzes werden als TTCN-3-Code erzeugt. Anschließend werden die Testfälle auf dem iXtronics TestRig automatisch ausgeführt. Die Testergebnisse werden wieder mit JUMBL analysiert.

Mit Hilfe des „**Time Partitioned Testing** (TPT)“ Ansatzes werden Testfälle über Zustandsautomaten mit variablem Verhalten definiert. Jeder Zustand kann unterschiedliche Verhaltensstränge referenzieren. Damit beschreibt ein einzelner Zustandsautomat eine Menge von Testfällen die sich aus der Permutation aller möglichen Verhaltensstränge der einzelnen Zustände ergibt. Die Wahl der Testfälle ist eine Entwicklungsaufgabe, wobei die Strukturierung der Testfälle, deren Modellierung über Zustandsautomaten und auch die Verarbeitung der Testergebnisse bestens durch TPT unterstützt wird. Die in D-MINT entstandene Integration von TPT in CAMeView/TestRig erlaubt die Ausführung der Testfälle und die Analyse der während eines Testfalles aufgezeichneten Messwerte.

Schließlich wird mit dem Werkzeug TTModeler das „**UML Testing Profile**“ in der Automatisierungsdomäne genutzt. Hier werden Testfälle manuell entwickelt und durch Sequenzmodelle dargestellt. Die Wahl der Testfälle ist auch hier eine Entwicklungsaufgabe wobei die Strukturierung und Weiterverarbeitung der Testfälle von dem Werkzeug unterstützt wird. Die Testfälle werden dann automatisch in TTCN-3 überführt und in der Testumgebung (iXtronics CAMelView/TestRig) ausgeführt.

### 12.1.3 Auswertung der Fallstudie

Die im Rahmen von D-MINT genutzten und an die Automatisierungsdomäne angepassten Methoden und Werkzeuge helfen der zukünftig ansteigenden Komplexität der Entwicklung von Geräten aus dieser Domäne zu begegnen.

Kategorie	Metrik	Verbesserung
Entwicklungsqualität	Aufwand einen Fehler zu finden und zu beheben.	20%
Testfallqualität	Wiederverwendung von Testfällen	10%
	Wiederverwendung von Modellen	80%
	Entwicklungsaufwand pro Testfall	0%
Aufwand	Manuelle Testfallausführung	50%
	Manuelle Ergebnisanalyse	50%

**Bild 12.3: Verbesserungen durch modellbasiertes Testen in der Automatisierungsdomäne**

In Bild 12.3 sind die Verbesserungen in drei Kategorien aufgeführt. Die Zahlenwerte basieren auf Messungen, existierenden Projekten und Schätzungen von Domänenexperten.

Der Aufwand, um einen Fehler zu finden und zu beheben konnte um 20% reduziert werden, was an den unterschiedlichen Wahrscheinlichkeitsverteilungen im Testmodell liegt, deren Veränderungen den Test unterschiedlicher Systemteile erlaubt, womit die Eingrenzung von Fehlern erleichtert wird. Die Wiederverwendung von Testfällen war und ist Teil der Entwicklungsarbeit. Die 10-prozentige Verbesserung basiert auf der verbesserten Wartbarkeit eines Testmodells im Vergleich zu manuell entwickelten Testfällen. Mit dem zentralen Testmodell des modellbasierten D-MINT Testansatzes ist die 80-prozentige Verbesserung bei der Wiederverwendung von Modellen einfach nachzuvollziehen. Dennoch werden existierende Testfälle weiter genutzt und durch die aus dem Testmodell generierten Testfälle ergänzt. Eine Verbesserung des Entwicklungsaufwandes pro Testfall konnte nicht verzeichnet werden, da die Einarbeitung in die neue modellbasierte Testmethode die Einsparungen ausglich. Nichtsdestotrotz gehen aktuelle Schätzungen davon aus, dass die Nutzung modellbasierter Testansätze nach dem dritten Entwicklungsprojekt eine Verbesserung des Entwicklungsaufwandes von 40% gegenüber herkömmlichen Testansätzen erreicht. Mit Hilfe der automatisierten Testfallausführung konnte die manuelle Testfallausführung um 50% reduziert werden. Schließlich wurde der Aufwand für die Analyse der Testfallergebnisse unter Nutzung der entsprechenden Werkzeuge um 50% reduziert.

Die gewichtete Zusammenfassung der Ergebnisse ergibt eine Verbesserung durch D-MINT Testansätze gegenüber nicht modellbasiertem Testen in der Automatisierungsdomäne um 35%. Diese Einsparung wird in

die Bewältigung der ansteigenden Komplexität zukünftiger Produkte investiert, um die Entwicklungszeit stabil halten und gleichzeitig mehr Kundenanforderungen in den Produkten realisieren zu können.

## 12.2 VORAUSSICHTLICHER NUTZEN

Die Projektergebnisse wurden zusammen mit der ABB Entwicklungsabteilung für Softstarter erarbeitet und werden dort auch für das Folgeprojekt genutzt. Allen weiteren Geschäftseinheiten stehen die Ergebnisse für zukünftige Projekte auf einer ABB internen Ergebnis-Transfer-Plattform zur Verfügung.

## 12.3 FORTSCHRITT BEI ANDEREN STELLEN

Die Resultate des modellbasierten Testens wurden in aufbereiteter Form auf einer ABB internen Ergebnis-Transfer-Plattform bereitgestellt, die allen Forschungszentren zu Verfügung steht. Damit ist ein wichtiger Meilenstein für die effiziente Verbreitung der D-MINT Ergebnisse in ABB gelegt. Darüber hinaus ist der Kontakt zu den an der Fallstudie aus der Automatisierungsdomäne beteiligten D-MINT Partnern (Fraunhofer IESE, iXtronics, Piketec und Testing Technologies) über das deutsche Forschungszentrum der ABB sichergestellt.

## 12.4 ERFOLGTE ODER GEPLANTE VERÖFFENTLICHUNGEN

Die Projektergebnisse sind, soweit nicht vertraulich, unter <http://www.d-mint.org> verfügbar. Weitere Veröffentlichungen sind:

- [94] Florian Kantz, Thomas Ruschival, Philipp Nenninger, Detlef Streitferdt, "Testing with Large Parameter Sets for the Development of Embedded Systems in the Automation Domain", in Proceedings of the 2nd International Workshop on Component-Based Design of Resource-Constrained Systems (CORCS 2009) at the 33rd Annual IEEE International Computers, Software and Applications Conference, Seattle, USA, 2009.
- [95] Thomas Ruschival, Philipp Nenninger, Florian Kantz, Detlef Streitferdt, "Test Case Mutation in Hybrid State Space for Reduction of No-Fault-Found Test Results in the Industrial Automation Domain", in Proceedings of the 2nd International Workshop on Industrial Experience in Embedded Systems Design (IEESD 2009) at the 33rd Annual IEEE International Computers, Software and Applications Conference, Seattle, USA, 2009.
- [96] Detlef Streitferdt, Philipp Nenninger, "Requirements on Testing in the Development of Embedded Automation Systems", in Proceedings of VDI Automation Kongress, Baden-Baden, 2008.
- [97] Detlef Streitferdt, Philipp Nenninger, Holger Kaul, Florian Kantz, "Low Power Development & Testing", in Proceedings of the Workshop on Low Power Design Impact on Test and Reliability at ETS08, 2008.
- [98] Detlef Streitferdt, Philipp Nenninger, Carlos Billich, Florian Kantz, Thomas Bauer, Robert Eschbach, "Model-based Testing in the Automation Domain, Safety Enabled", in Proceedings of the 1st Workshop on Model-based Testing in Practice, MoTiP at the Fourth European Conference on Model Driven ArchitectureFoundations and Applications (ECDMA), Berlin, 2008.
- [99] Florian Kantz, Detlef Streitferdt, and Philipp Nenninger, "Generic Interface for Modelbased Testing in theAutomation Domain", in Proceedings of the 1st Workshop on Model-based Testing in Practice, MoTiP at the ECDMA Conference, Berlin, 2008.
- [100] Detlef Streitferdt, Philipp Nenninger, Georg Wendt, ABB - Industrial Automation Case Study, D-MINT deliverables D1.1 / D1.2, 2007/2008
- [101] Detlef Streitferdt, Automation Domain Case Study Manual, D-MINT deliverable D1.3, 2008
- [102] Detlef Streitferdt, Philipp Nenninger, Automation Domain Case Study Final Requirements, D-MINT deliverable D1.4, 2008
- [103] Detlef Streitferdt, ABB Automation case Study Evaluation, D-MINT deliverable D1.5, 2009
- [104] D-MINT consortium: D-MINT white paper, Common approach to architecture driven testing, D-MINT deliverable CA\_WP\_v1.6, [http://www.d-mint.org/public/CommonApproach\\_WhitePaper\\_DMINT.pdf](http://www.d-mint.org/public/CommonApproach_WhitePaper_DMINT.pdf)

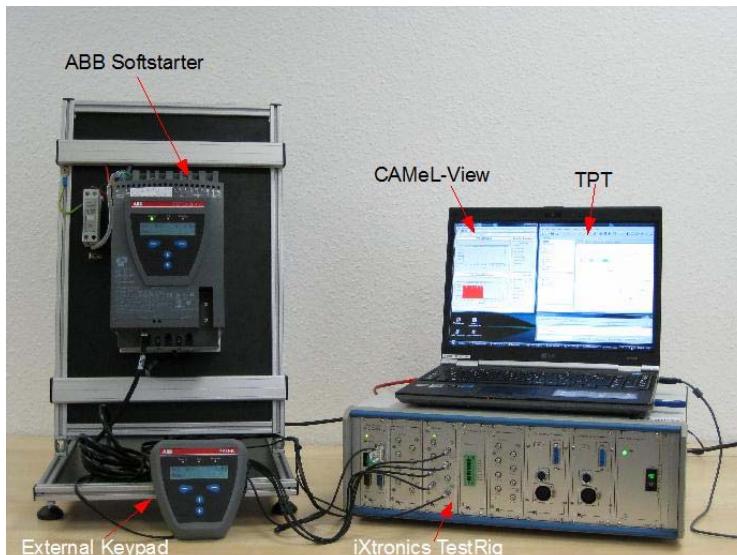
## 13. IXTRONICS GMBH, FÖRDERKENNZEICHEN 01 IS 07 001 H

iXtronics hat im Rahmen des Projektes als Tool Provider in der ABB Fallstudie gearbeitet. Für die Fallstudie wurden Software und Hardwarekomponenten bereitgestellt und für die Integration und Kopplung mit den Werkzeugen der weiteren Tool Provider wesentlich weiterentwickelt.

### 13.1 ERREICHTE ERGEBNISSE

An der Fallstudie des ABB Softstarters wurde gezeigt, wie echtzeitfähige Testmodelle für ein SUT in Kombination mit den Software Werkzeugen von PikeTec, Testing Tec und Fraunhofer IESE entwickelt werden können. Von iXtronics wurde dazu die Software CAMeL-View und die Echtzeithardware CAMeL-View TestRig bereitgestellt und auf die projektspezifischen Anforderungen erweitert.

Für die Präsentation der Ergebnisse auf Messen und bei potentiellen Kunden bzw. Anwendern wurde zusätzlich zum Demonstrator bei ABB, auch bei iXtronics ein Demonstrator aufgebaut. Der Demonstrator besteht aus einem Softstarter und einer TestRig Hardware (siehe Bild 13.1).



**Bild 13.1: Demonstratoraufbau bei iXtronics**

Der Demonstrator diente auch zur parallelen Entwicklung der Testkomponenten und Schnittstellen zwischen den im folgenden beschriebenen Werkzeugen.

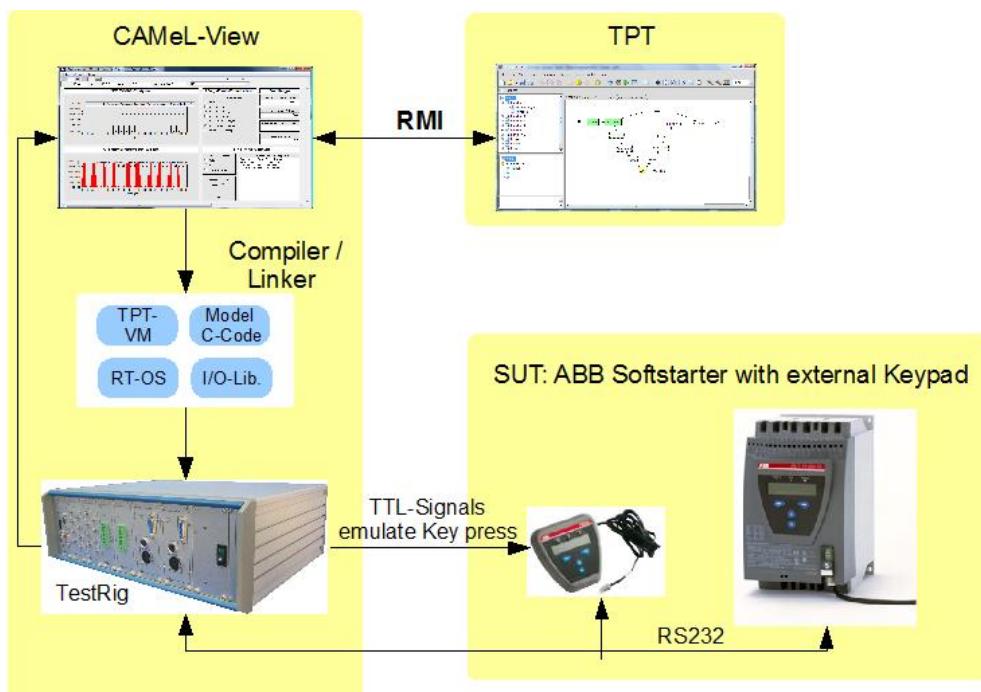
#### 13.1.1 Werkzeugkopplung mit PikeTec TPT

Im D-MINT Projekt wurde eine erste Implementierung einer Werkzeug-Kopplung/-Integration mit der Software TPT (Time Partition Testing) von PikeTec realisiert. Bild 13.2 zeigt die Struktur der Kopplung. Die Beschreibung des Testmodells bzw. der Testabläufe für das SUT wird in der Software TPT realisiert. Die Beschreibung des echtzeitfähigen Testmodells und die physikalische Ankopplung des SUT erfolgt in CAMeL-View.

Um die Tests in Echtzeit verarbeiten zu können, wurde die virtuelle Maschine von TPT in CAMeL-View integriert. In TPT wird der Testablauf für das SUT beschrieben. Aus dieser Testbeschreibung wird ein Bytecode für die virtuelle Maschine generiert. Der Bytecode wird auf die virtuelle Maschine geladen und dann zusammen mit dem CAMeL-View Testmodell des SUT in Echtzeit auf der TestRig Hardware ausgeführt. Die Kommunikation zwischen TPT und CAMeL-View während des Testablaufs geschieht in einer niederprioren Backgroundtask. So kann die Verarbeitung des Tests in Echtzeit ablaufen und gleichzeitig auch eine Visualisierung der ermittelten Testdaten stattfinden.

Über eine RMI (Remote Message Invocation) Schnittstelle, die im Rahmen des D-Mint Projektes entwickelt wurde, erfolgt die Steuerung der CAMeL-View Software. Das realisierte Interface erlaubt folgende Interaktionen zwischen TPT und CAMeL-View:

- Laden und Starten von Modellen unter CAMeL-View
- Setzen und Lesen von Modellparametern
- Zugriff auf die Daten der virtuellen Maschine von TPT in CAMeL-View
- Lesen der Testergebnisse

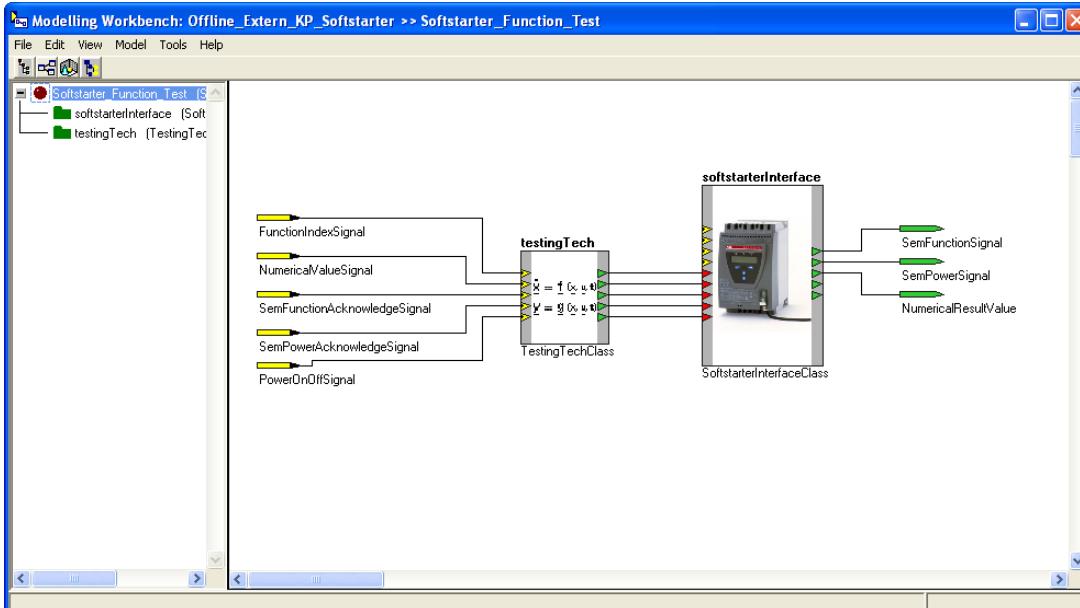


**Bild 13.2: Testablauf mit TPT und CAMeL-View TestRig**

### 13.1.2 Werkzeugkopplung mit TestingTech TTWorkbench

Parallel zu den Arbeiten mit der Fa. PikeTec wurde auch eine Kopplung mit dem Softwarewerkzeug TTWorkbench von TestingTec realisiert (siehe Bild 13.3). Als Schnittstelle zur Kommunikation zwischen TTWorkbench und CAMeL-View wird auch hier die RMI-Schnittstelle verwendet.

Im Unterschied zur Lösung mit TPT geschieht die Abarbeitung der Testabläufe in einem schrittweise gesteuerten Prozess mit dem Senden eines Testbefehls und dem Warten auf eine Antwort vom SUT. In TTWorkbench wird dazu der Testablauf als Sequenz von Testbefehlen beschrieben. Auf der Ausführungsseite im CAMeL-View Testmodell für das SUT wird dieser Befehlsumruf in entsprechende Befehlsequenzen zur Ansteuerung des SUT umgesetzt. Diese Befehlsequenzen werden in Echtzeit verarbeitet. Ein Befehlsumruf aus der TTWorkbench kann einem einzelnen Befehl im CAMeL-View Testmodell entsprechen oder aus einer Sequenz von Befehlen bestehen. Die Antwort des SUT wird via dem Testmodell an die TTWorkbench gesendet.



**Bild 13.3: CAMEL-View SUT Testmodell für Kopplung mit TTWorkbench**

Das realisierte Interface zwischen TTWorkbensch und CAMEL-View erlaubt folgende Interaktionen:

- Laden und Starten von Modellen unter CAMEL-View
- Setzen und Lesen von Modellparametern
- Lesen der Testergebnisse

### 13.1.3 Fallstudie ABB Softstarter

Die zuvor beschriebenen Werkzeugkopplungen und deren Funktionalität zum Modellbasierten Testen wurden am Beispiel das ABB Softstarters erfolgreich getestet.

Es wurde ein Testmodell zur physikalischen Kopplung des Softstarters mit dem CAMEL-View TestRig System erstellt. Das Testmodell erlaubt es mittels TTL-Signalen die Tastendrücke des externen Softstarter Keypads zu emulieren. Die Antwort des Softstarters auf diese simulierten Tastendrücke werden via RS232 Schnittstelle zum TestRig kommuniziert. Die Testsequenzen wurde in TPT und TTWorkbench erstellt.

In der Fallstudie wurden mehrere Tests definiert. Ein Test bestand z. B. in der Umschaltung des Sprachmenüs des Softstarters. Durch die simulierten Tastendrücke wurden die verschiedenen Menüstrukturen durchlaufen, die eine Umschaltung des Displaymenüs von deutscher Sprache auf z. B. Spanisch vollzog und wieder zurück zur deutschen Menüsprache.

Ein weiterer Test bestand darin die Rampen für das Anlaufen und Stoppen eines Motors mit dem Softstarter zu konfigurieren und danach den Softstarter für eine bestimmte Zeit auszuschalten. Danach den Softstarter wieder einzuschalten und die zuvor programmierten Parameter wieder einzulesen und mit den Vorgabewerten zu vergleichen. Dieses Beispiel wurde auf dem ITEA Symposium im Madrid als eins der Anwendungsbeispiele demonstriert.

 Deployment of Model-Based Technologies to Industrial Testing	<b>D-MINT Abschlussbericht</b>	Seite : 60 of 81 Version: 1.0 Datum: 30.06.2010 <hr/> Status : Final
--	--------------------------------	---

### 13.2 VORAUSSICHTLICHER NUTZEN

Die bis heute gewonnenen Teilergebnisse und Erkenntnisse werden von iXtronics bereits mit Kunden aus anderen Projekten diskutiert, um etwaige Anwendungsfälle frühzeitig absehen und den jeweiligen Bedarf abschätzen zu können. Insbesondere das automatisierte Testen und Überprüfen von Modellen spielt hier eine große Rolle. Dies betrifft neben der Automobilindustrie insbesondere auch die Luftfahrtindustrie und die Feinwerktechnik.

Es ist eine weitergehende Kooperation mit der Fa. PikeTec geplant. Ziel dieser Kooperation soll die gemeinsame Verwertung der Ergebnisse der Werkzeugkopplung zwischen CAMeL-View und TPT sein.

### 13.3 FORTSCHRITT BEI ANDEREN STELLEN

Durch die Arbeiten an D-Mint ist das Interesse an den iXtronics Lösungen für den modellbasierten Entwurf mechatronischer Systeme insgesamt gestiegen und hat zu ersten Projektgesprächen z. B. in der Feinwerktechnik geführt.

### 13.4 ERFOLGTE ODER GEPLANTE VERÖFFENTLICHUNGEN

Die Projektergebnisse sind, soweit nicht vertraulich, unter <http://www.d-mint.org> verfügbar. Weitere Veröffentlichungen sind:

- [105] Präsentation auf der Embedded World 2008 in Nürnberg, Feb. 2008.
- [106] Präsentation auf der SPS IPC DRIVES 2008 in Nürnberg. Vorgestellt wurde die Toolkopplung zwischen der iXtronics Software CAMeL-View und der PikeTec Software TPT am Beispiel des ABB Softstarters. Nov. 2008.
- [107] Offener Workshop CAMeL-View Essentials in Paderborn; Vermittlung von Techniken zum modellbasierten Testen, Juni 2009.
- [108] Präsentation auf dem 6. Paderborner Workshop „Entwurf mechatronischer Systeme“. 2-3.4.2009, Heinz-Nixdorf Forum Paderborn, Begleitende Fachausstellung.

## 14. INSPIRE AG, FÖRDERKENNZEICHEN 01 IS 07 001 I

### 14.1 ERREICHTE ERGEBNISSE

In die Arbeiten des Arbeitspaketes 4 – Verwertung waren alle D-MINT Partner involviert. Daimler, iXtronics, TestingTech, IESE und INSPIRE arbeiteten in diesem Arbeitspaket an allen Tasks, die D-MINT für eine erfolgreiche und professionelle Verwertung benötigt: Verwertungsplanung (14.1.1), Produktpositionierung (14.1.2), industrielle Metriken (14.1.3).

Die Verwertungsmöglichkeiten in einem Projekt wie D-MINT hängen sehr stark von der Rolle und dem Fokus der beteiligten Organisation ab. Daher müssen die Verwertungsmöglichkeiten explizit ausgearbeitet werden, um tatsächlich umgesetzt werden zu können.

Die Bereiche der Verwertung wurden aus diesem Grund in die folgenden drei Kategorien aufgeteilt:

1. Externe Verwertung
2. Interne Verwertung
3. Lösungen mit Produktpotential

#### 14.1.1 Verwertungsplanung

##### a) Ziele

Das Ziel dieses Tasks war es, die Verwertung der Projektergebnisse zu bündeln und zu planen. Hierzu diente in erster Linie der Verwertungsplan mit den Beiträgen der Projektpartner, der über die Projektlaufzeit fortgeschrieben wurde.

Darüber hinaus hat das Verwertungsteam innerhalb dieses Tasks verschiedene Indikatoren gemessen und bewertet, die den Erfolg des Projektes darstellen. Dies waren zum einen die Messung der entwickelten Innovationen, die Messung der Umsetzung des Verwertungsplans und die Berechnung des Return on Investment für das Projekt.

##### b) Verwertungsplan

Die Verwertung der Projektergebnisse wurde zur besseren Transparenz in drei Bereiche aufgeteilt, die jeweils maßgeblich einer Partnerkategorie zugeordnet werden kann.

##### Externe Verwertung

Externe Verwertung beschäftigt sich mit den Marktbedürfnissen, Chancen und Möglichkeiten, die Marktposition eines D-MINT Partners zu verbessern. Dies kann durch neue oder verbesserte Softwareprodukte im Falle eines Softwareherstellers geschehen, aber auch durch die Entwicklung neuer Services wie zum Beispiel Consulting. Bei Forschungsinstituten kann hierunter aber unter anderem auch das Angebot von Seminaren oder Konferenzen oder die Veröffentlichung von Publikationen gefasst werden.

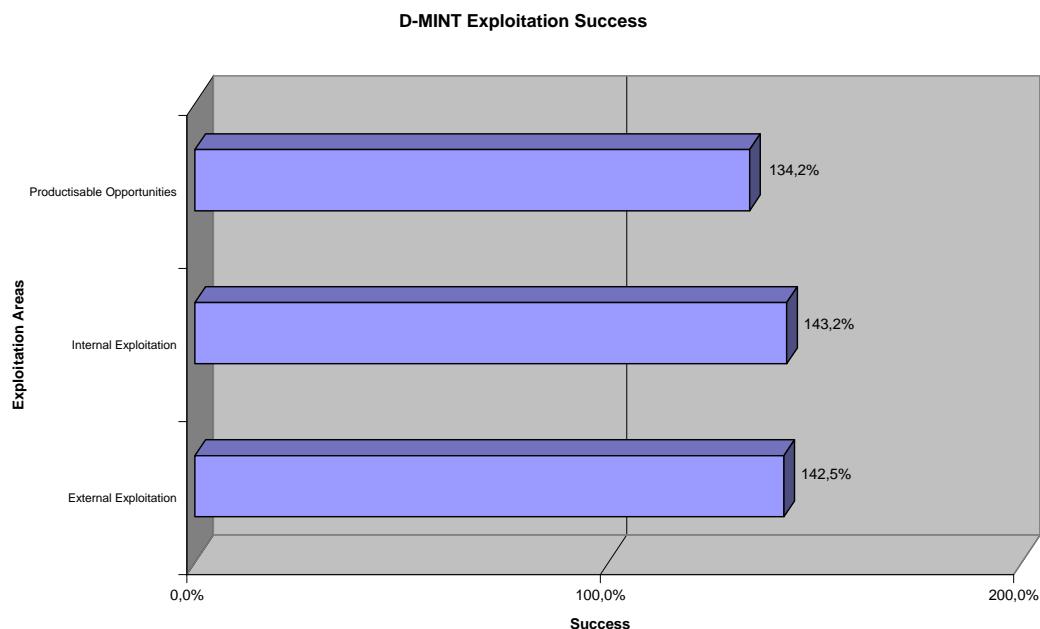
##### Interne Verwertung

Interne Verwertung richtet sich mehr auf die organisationsinterne Nutzung und Anwendung der Forschungsergebnisse und deren Einfluss auf Effizienzsteigerungen. Hierunter fällt zum Beispiel die Verbesserung interner Prozesse zur Kostensenkung oder zur Qualitätsverbesserung. Auch die Verkürzung der Produktentwicklungszeiten spielt hier eine Rolle. Im D-MINT Projekt wurde interne Verwertung vorwiegend durch die Durchführung der Fallstudien und die Entwicklung der Demonstratoren betrieben. Die entwickelten Ansätze und Methoden können sowohl die Effizienz der Produkttests deutlich erhöhen als auch zur Fehlervermeidung beitragen, indem die Fehlerfälle früher in der Entwicklungsphase gefunden werden.

##### c) Verwertungserfolg

Da die Konsortiumsmitglieder für ihre individuellen Organisationen Ziele im Bezug auf die Verwertung definiert haben, wollte das Verwertungsteam wissen, in welchem Umfang diese Ziele realisiert werden

können und hat daher die Einschätzungen der Partner aufgenommen. Das Ergebnis der Befragung ist in der folgenden Abbildung zu sehen:



**Bild 14.1: D-MINT Exploitation Success**

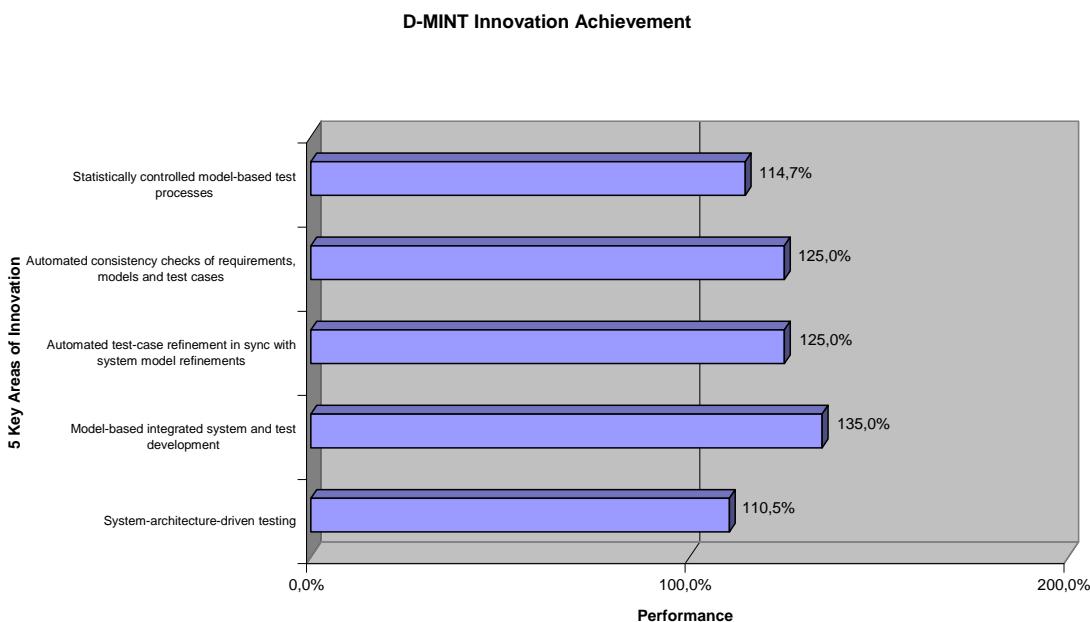
Hier wird deutlich, dass die Verwertungspläne in hohem Maße realisiert werden dürfen, da die Erwartungen weit im positiven Bereich liegen. Dies zeigt, dass die Erwartungen, die das Konsortium zu Beginn des Vorhabens definiert hat, sehr realistisch waren. Es zeigt aber auch die Anstrengungen, die die Partner in das Projekt investiert haben.

#### **d) Innovation**

Zu Beginn des Projektes hatte sich das Konsortium Innovationen in fünf Schlüsselbereichen zum Ziel gesetzt. Das Verwertungsteam hat analysiert, in welchem Ausmaß diese Innovationen tatsächlich realisiert wurden. Aus diesem Grund wurden die Einschätzungen der Partner zu diesem Thema aufgenommen und für die jeweiligen Bereiche beziffert. Die Innovationsfelder sind hier nochmals beschrieben:

1. Architekturgetriebenes Testen: Ein neuer Ansatz für das modellbasierte Testen, bei dem die Systemarchitektur als Integrator für heterogene Komponentenmodelle dient und bei denen Testmuster als Richtlinie für die Testfallerzeugung für den industriellen Einsatz dienen.
2. Modellbasierte System-Integration und Test-Entwicklung: Entwicklung von Konzepten für das konsistente Management von Anforderungen und von Testfällen für verteilte und kompositionelle System-Strukturen.
3. Automatisierte Testfallverfeinerung in Synchronisation mit der Systemmodellverfeinerung: Entwicklung eines automatisierten Konzepts, um Testfälle mit der Verfeinerung von Entwurfsmodellen Schritt halten zu lassen.
4. Automatisierte Konsistenzprüfungen von Anforderungen, Modellen und Testfällen bzw. Testmodellen (bzw. Teilen hiervon) soll nicht nur syntaktisch, sondern auch semantisch in automatisierter Form unterstützt werden.

5. Statistisch gesteuerter modellbasierter Testprozess: Statistische Methoden sollen für den modellbasierten Testprozess verwendet werden, um den Prozess zu messen, zu kontrollieren und um die Prozess- und Systemeigenschaften zu verbessern.



**Bild 14.2: D-MINT Innovation Achievement**

Die Abbildung zeigt, dass das Projekt in allen angestrebten Innovationsbereichen über den ursprünglichen Erwartungen liegt, was unter Berücksichtigung der Komplexität des Vorhabens ein sehr gutes Ergebnis ist.

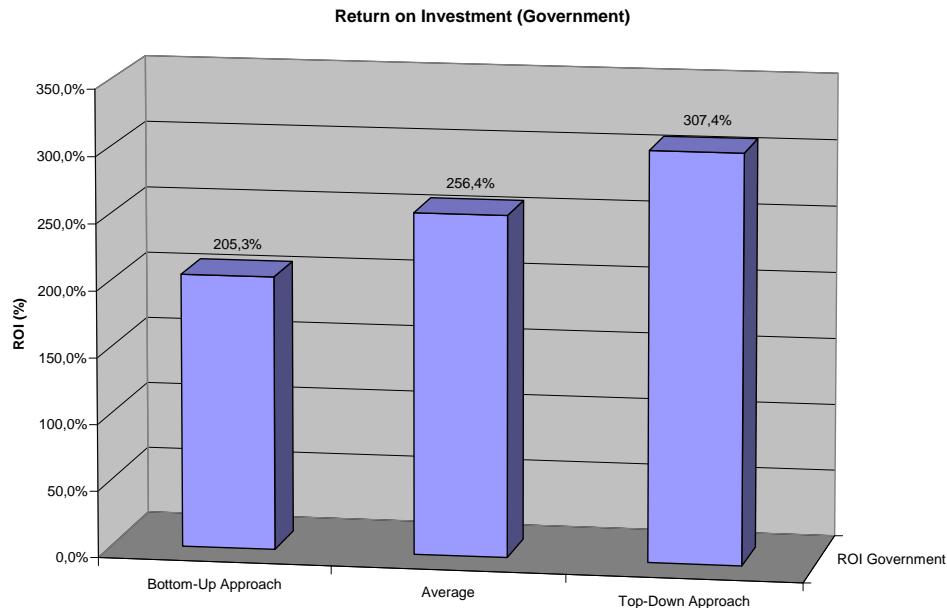
### e) *Return on Investment*

Der Return on Investment zeigt am deutlichsten den Erfolg des Projektes aus einer monetären Perspektive. Das Verwertungsteam hat hierbei sowohl einen Top-Down als auch einen Bottom-Up Ansatz verfolgt, um ein realistisches Bild der Situation zu bekommen. Aufgrund von Geheimhaltungsvereinbarungen dürfen hier keine Einzeldaten veröffentlicht werden.

Für den Bottom-Up Ansatz hat das Verwertungsteam vielfältige Daten von den Partnern gesammelt, um die Kosten und den Nutzen des Projektes zu berechnen. Der Nutzen für Softwarehersteller bezog sich dabei im Wesentlichen auf externe Umsätze und der für Transferpartner besonders auf Effizienzsteigerungen.

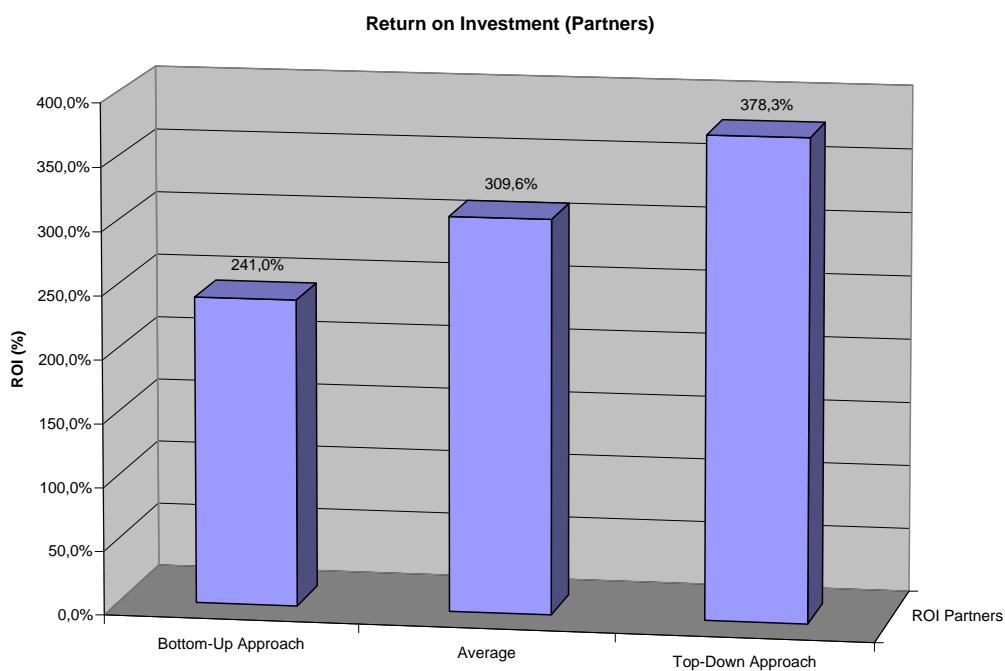
Der Top-Down Ansatz berücksichtigt alle Nutzenaspekte, die das Konsortium als ganzes durch das Projekt erzielt hat. Die erzielten Resultate wie Softwareprodukte, Seminare und Publikationen wurden monetär und mengenmäßig bewertet. Auch die Aufwendungen wurden auf der Basis des Konsortiums berechnet.

Der ROI wird zum einen für den Kapitaleinsatz der Partner berechnet, zum anderen für den Kapitaleinsatz der offiziellen Stellen, die das Funding bereitgestellt haben. Diese werden als Partner ROI und als Government ROI bezeichnet.



**Bild 14.3: Return on Investment (Government)**

Ein Wert von 256 % für den Government ROI als Mittelwert zwischen den beiden Ansätzen ist ein sehr gutes Resultat, da dies bedeutet, dass aus jedem Euro, der investiert wurde, 2,56 € Rückfluss erzielt werden können. Der Partner ROI liegt aufgrund der Förderquote etwas höher.



**Bild 14.4: Return on Investment (Partners)**

Die ROI-Zahlen zeigen sehr deutlich, dass das D-MINT Projekt sowohl für die Partner als auch für die Förderer substanzelle Vorteile erwirtschaften kann, sogar wenn ausschließlich die monetäre Situation betrachtet wird. Darüber hinaus sind aber auch nennenswerte nicht-monetäre Vorteile wie Know-how-

 <b>D-MINT</b> <small>Deployment of Model-Based Technologies to Industrial Testing</small>	<b>D-MINT Abschlussbericht</b>	Seite : 65 of 81 Version: 1.0 Datum: 30.06.2010 Status : Final
---	--------------------------------	---

Gewinn und Erfahrung in der praktischen Anwendung von modellbasiertem Testen – besonders im Bereich der Forschungsinstitute – erzielt worden.

### **f) Fast Exploitation – Verwertung während des Projektes**

Bereits innerhalb des Projektes sind einige Verwertungserfolge zu verzeichnen. Diese sind hier kurz aufgelistet:

Nutzung von D-MINT Technologie durch die Transferpartner:

- ELIKO nutzt D-MINT Tools and Technologien für Produkte, die bereits verkauft werden und sich auf dem Markt im Einsatz befinden
- ABB hat entschieden, D-MINT Resultate für die nächste Version von Softstarter zu nutzen
- Daimler hat sich entschieden, D-MINT Technologien für ECU-Software Testing zu benutzen

Softwarehersteller, die ihre Produkte bereits auf den Markt gebracht haben:

- iXtronics Toolbox
- Testing Tech TTmodeler
- Conformiq Qtronic

## **14.1.2 Produktpositionierung**

### **a) Ziele**

Der Task 4.2 beschäftigte sich mit der Positionierung möglicher Produkte aus dem Projekt. Für Lösungen mit Produktpotential wurde eine Analyse und Studie durchgeführt, um Marktpotential und Wettbewerb zu untersuchen. Diese Lösungen wurden in einem separaten Dokument dargestellt und bewertet. Dabei handelte es sich im Wesentlichen um Weiterentwicklungen bereits bestehender Produkte und die Integration von D-MINT Technologie in diese Softwaresysteme.

### **b) Marktsituation**

Der Markt für Testdienstleistung wird nach den Erwartungen des Software Consulting Unternehmens Ovum (März 2009) bis 2013 auf ein Volumen von 56 Milliarden US\$ anwachsen. Damit wird die Wachstumsrate im Mittel über die Jahre 2008 bis 2013 bei 9,5 % liegen, was deutlich über den anderen Dienstleistungsbereichen im IT Sektor liegt.

Im gleichen Zeitraum hat Gartner Research, der weltgrößte Research-Dienstleister im IT-Markt, seinen Bericht 'Market Trends: Application Development, Worldwide 2008-2013' veröffentlicht. Danach hat der weltweite Testmarkt für Software ein Volumen von 13 Milliarden US\$ und der Markt für Outsourcing in diesem Bereich ein Volumen von 6,1 Milliarden US\$.

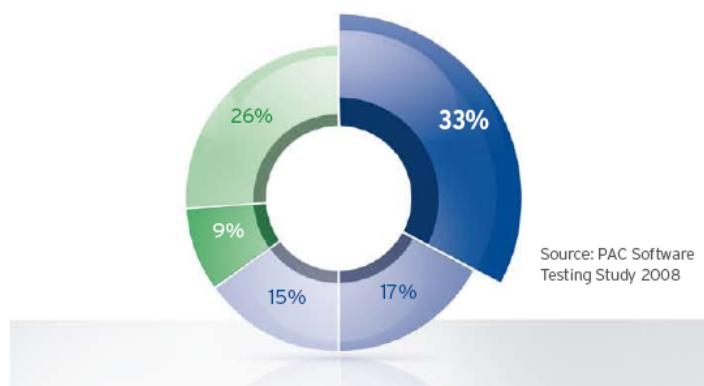
Nach einer Studie, die kürzlich in Indien durchgeführt wurde, erreicht Software Testing einen Anteil von 24 % an allen Einstellungen im IT-Bereich. Damit liegt Software Testing an zweiter Stelle nach den Einstellungen in Callcentern, die bei 47 % liegen.

Diese Zahlen verdeutlichen, dass der Testmarkt, in dem das D-MINT Projekt angesiedelt ist, ein hohes Wachstumspotential hat, in dem Europa und Deutschland sich gegen hochrangige Konkurrenten werden behaupten müssen.

Dieses Kapitel bietet einen Überblick über den Testmarkt und wird untermauert durch eine Marktstudie mit 842 Unternehmen aus Deutschland, der Schweiz und Österreich. Diese Studie wurde im Auftrag von SQS durchgeführt, die im Konsortium des D-MINT Projektes vertreten sind. Die folgenden Fragen stellen eine wichtige Grundlage für die Bewertung der entwickelten Lösungen dar.

### Wie wichtig ist Testen für Ihr Unternehmen?

- Testing is a vital investment in the product
- Testing raises the cost efficiency of software production
- Test tools create true added value in the company
- Testing is a necessary evil
- Other/no response

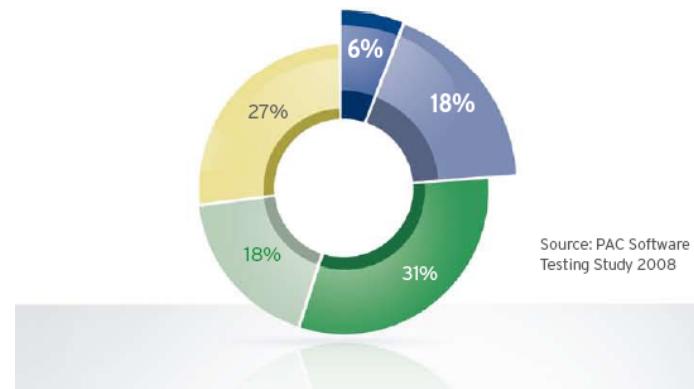


**Bild 14.5: Wichtigkeit des Testens (Quelle: PAC Software Testing Study 2008)**

Die Abbildung zeigt deutlich, dass die meisten Unternehmen im IT-Markt die Relevanz von Tests als wichtigen Bestandteil der Produktentwicklung sehen. Dies dürfte auch für Unternehmen aus anderen Märkten zutreffen.

### Wie viele ausgelieferte Applikationen hatten einen negativen Einfluss auf den Kunden?

- Most
- Some
- Few
- None
- No response

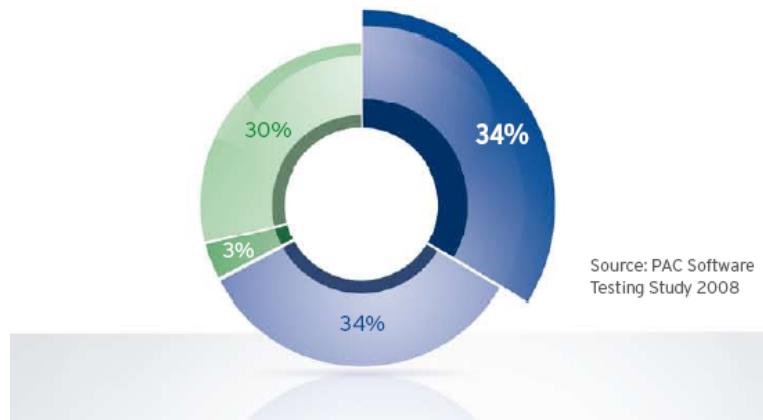


**Bild 14.6: Fehler in Software Applikationen (Quelle: PAC Software Testing Study 2008)**

Es zeigt sich, dass im Testmarkt ein Zielkonflikt zwischen einer schnellen Time-to-Market und der Fehlerfreiheit des Produktes besteht. Dabei ist allerdings eine fehlerhafte Software ein großes Risiko für das zukünftige Geschäft. Diese Sichtweise kann auch aus den Untersuchungen innerhalb des D-MINT Projektes abgeleitet werden.

### Wird Ihr Budget für externes Testen wachsen?

- Will grow
- Will stay the same
- Will shrink
- Don't know/no response

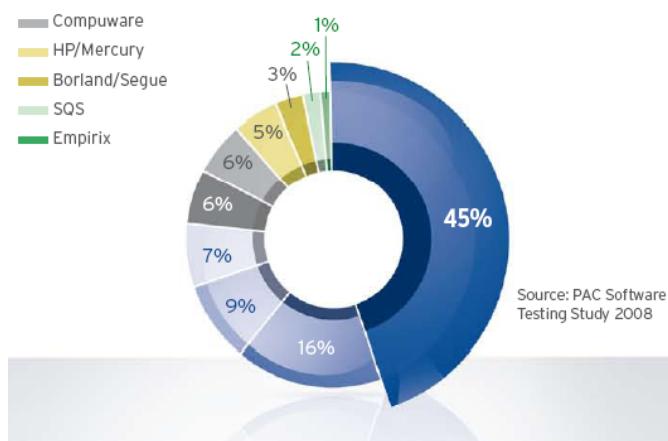


**Bild 14.7: Budget für Testen (Quelle: PAC Software Testing Study 2008)**

Bisher ist der Testmarkt eher ein Markt für interne Dienstleister, da der Großteil der Testaktivitäten in den Unternehmen selbst stattfindet. Da der externe Anteil aber wächst, haben Dienstleister, die diesen Markt adressieren, ein gutes Wachstumspotential, sofern sie die Bedürfnisse der Kunden verstehen und Lösungen für dessen Probleme finden.

### Welche Testsoftware setzen Sie in Ihrem Unternehmen ein?

- Other
- Don't know/no response
- Open Source
- Excel
- IBM Rational
- Compuware
- HP/Mercury
- Borland/Segue
- SQS
- Empirix



**Bild 14.8: Testing Tools (Source: PAC Software Testing Study 2008)**

 <b>D-MINT</b> <small>Deployment of Model-Based Technologies to Industrial Testing</small>	<b>D-MINT Abschlussbericht</b>	Seite : 68 of 81 Version: 1.0 Datum: 30.06.2010 <hr/> Status : Final
---	--------------------------------	---

Aus den Ergebnissen kann geschlossen werden, dass der Testmarkt eine Phase der Standardisierung durchlaufen wird, wie dies in vielen anderen Softwarebereichen bereits geschehen ist. Diese Chance kann durch Unternehmen mit guten Konzepten und Methoden genutzt werden, um den Markt zu durchdringen, wenngleich diese Phase sehr von Wettbewerb geprägt sein wird und sicher auch eine Konzentrationsphase nach sich ziehen wird.

### c) *Lösungen mit Produktpotential*

Das Verwertungsteam hat aufgrund der Datenlage beschlossen, bei der Auswahl der Lösungen die mit deutscher Beteiligung durchgeführten Fallstudien besonders zu berücksichtigen.

Diese Fallstudien sind hier aufgelistet:

- Automotive Fallstudie unter der Leitung von Daimler
- Telecommunication Fallstudie unter der Leitung von NSN
- Industrial Automation Fallstudie unter der Leitung von ABB

In der Darstellung dieser Lösungen wurde zunächst eine Produktbeschreibung vorgestellt, der Hauptaugenmerk lag aber in der Untersuchung der folgenden Punkte als den wesentlichen Marktgegebenheiten:

- Vorteile für potentielle Kunden
- Wettbewerbsvorteile (Unique Selling Proposition)
- Marktsituation und Wettbewerb
- Marketingstrategie

Folgende Produkte wurden im Zuge der Untersuchung ausgewählt, wobei die letzten beiden genannten nicht aus den oben genannten Fallstudien stammen:

- Conformiq Qtronic
- iXtronics CAMeL-View
- Nethawk EAST
- PikeTec TPT
- Testing Tech TTworkbench and TTmodeler
- Elvior MessageMagic
- Elvior MOTES

Unter der folgenden Gliederung finden sich Teile der oben genannten Lösungen ebenfalls wieder, sind an dieser Stelle aber nach ihrem Einsatzgebiet gegliedert.

#### MBT Lösungen für Embedded Systems

- PikeTec TPT / iXtronics TestRig
- FOKUS ADT

#### MBT Lösungen für Reaktive Systeme

- Elvior MOTES

#### MBT Lösungen für Telekommunikationssysteme

- Abo MATERA / Conformiq Qtronic
- Conformiq Qtronic / Nethawk EAST

Es hat sich gezeigt, dass innerhalb des D-MINT Projektes zahlreiche zukunftsfähige Lösungen entwickelt wurden, die alle Marktpotential besitzen beziehungsweise durch das Projekt eine Weiterentwicklung erfahren haben, sofern sie bereits vorher auf dem Markt waren.

Wenn die investierte Arbeit weiterverfolgt wird, entweder in den jeweiligen Unternehmen oder in Rahmen von Folgeprojekten, ist es sehr wahrscheinlich, dass die D-MINT Technologie für Model Based Testing in Zukunft weiteren Industrien zur Verfügung stehen wird. Die Vorteile von MBT liegen auf der Hand und konnten durch die Fallstudien auch unter Beweis gestellt werden. Durch die Verbreitungsaktivitäten wird dazu beigetragen, dass das im Projekt gesammelte Wissen weitere Verbreitung findet.

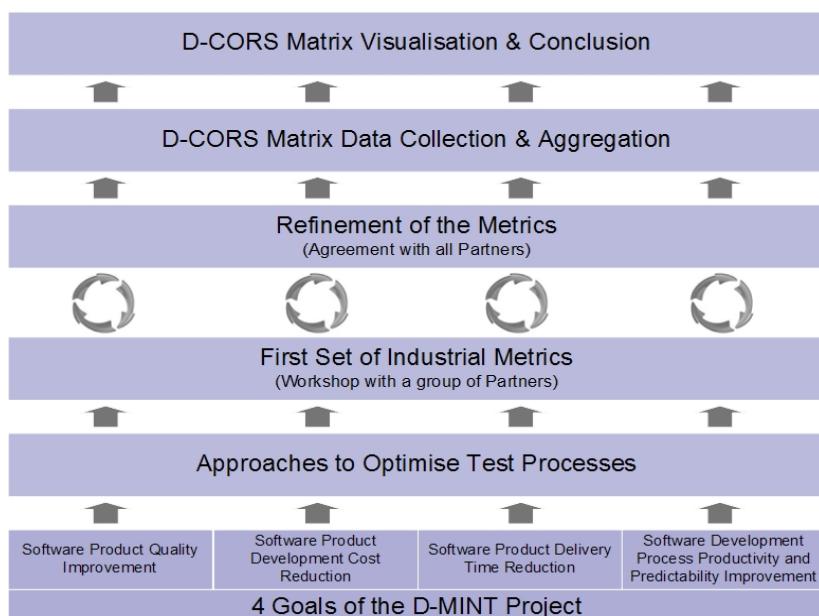
### 14.1.3 Industrielle Metriken

### a) Ziele

Innerhalb des Tasks 4.3 wurde ein System von Metriken entwickelt, mit dessen Hilfe die Ergebnisse aus den Fallstudien und Demonstratoren bewertet werden können. Das Ziel dieser Bewertung war es, über die unterschiedlichen Industrien hinweg, aber auch in einer Gesamtsicht die relevanten Kriterien zu ermitteln, die für Unternehmen ausschlaggebend für die Entscheidungen im Bezug auf Testmethoden und -verfahren sind.

Diese Kriterien dienen dazu, die Softwarehersteller im Konsortium bei der Ausrichtung auf den Markt zu unterstützen und gleichzeitig durch die Messung dieser Indikatoren die Leistungsfähigkeit der im Projekt erarbeiteten Methoden und Technologien unter Beweis zu stellen.

### **b) Metriken**



**Bild 14.9:** Entwicklung der Metriken

In Task 4.3 wurde basierend auf dem im Schaubild beschriebenen Ansatz ein System von Indikatoren zur Bewertung von Testprozessen entwickelt, das hier beschrieben wird.

Testqualität	
1	<b>Testredundanz</b> Diese Metrik misst die Vermeidung oder Reduzierung von Redundanz der Testfälle in verschiedenen Szenarien. Testfälle sollen nur einmalig durchgeführt werden, zumindest nicht in mehreren Bereichen, sondern auf koordinierte Weise. Dies wird durch Modelle möglich.

2	<b>Abdeckung der Anforderungen</b> <p>Es ist wichtig, dass alle Anforderungen, die extern oder intern definiert wurden, zuverlässig geprüft und im Design abgedeckt werden. Nur dann kann ein zukünftiges Produkt die Erwartungen der Kunden erfüllen. Die Anforderungen sollen zu 100 % abgedeckt werden, ohne jedoch Redundanz zu erzeugen.</p>
3	<b>Testdauer</b> <p>Um eine schnelle Time-to-Market zu erzielen, ist es besonders wichtig, die Testdauer auf ein Minimum zu reduzieren, ohne die Qualitätsaspekte aus den Augen zu verlieren. Die ersten drei Metriken sind tatsächlich unmöglich zeitgleich zu minimieren, sondern es muss ein Optimum durch jedes Unternehmen gefunden werden.</p>
4	<b>Fehlerdiagnosefähigkeit</b> <ul style="list-style-type: none"> <li>- <b>Anzahl der Fehler</b></li> <li>- <b>Anzahl der kritischen Fehler</b></li> <li>- <b>Anzahl der wiederkehrenden Fehler</b></li> </ul> <p>Die Fehlerdiagnosefähigkeit wird durch die drei Unterpunkte definiert. Das Ziel ist es, möglichst viele Fehler möglichst früh in der Entwicklung zu finden. Besonders die kritischen und wiederkehrenden Fehler sollen komplett ausgemerzt werden, bevor das Produkt auf den Markt kommt.</p>

Kosten	
5	<b>Personalkosten [intern und extern]</b> <ul style="list-style-type: none"> <li>- <b>Training der Mitarbeiter</b></li> <li>- <b>Design des Testmodells</b></li> <li>- <b>Entwicklung der Testfälle</b></li> <li>- <b>Qualitätssicherung &amp; Anpassung der Testfälle</b></li> </ul> <p>Personalkosten sind ein wesentlicher Bestandteil der Gesamtkosten eines Testprojektes. Diese können in diese vier Kategorien unterteilt werden. Das Ziel der Unternehmen ist es, die Personalkosten zu senken, ohne die Qualität zu vernachlässigen.</p>
6	<b>Materialkosten</b> <ul style="list-style-type: none"> <li>- <b>Hardware</b></li> <li>- <b>Software</b></li> </ul> <p>In den meisten Fällen spielen Materialkosten nur eine untergeordnete Rolle bei der Betrachtung von Testprojekten. Da es aber bestimmte Bereiche gibt, in denen die Komplexität der Produkte zu signifikanten Materialkosten führt, wurde diese Metrik aufgenommen.</p>
7	<b>Fehlerbehebungskosten</b> <ul style="list-style-type: none"> <li>- <b>in der Entwicklungsphase</b></li> <li>- <b>nach der Auslieferung</b> <ul style="list-style-type: none"> <li>o <b>Gewährleistung</b></li> <li>o <b>PR für Imagekampagnen</b></li> <li>o <b>Schadenersatz</b></li> </ul> </li> </ul> <p>Die am schwierigsten einzuschätzenden Kosten bei Testprojekten sind die Fehlerbehebungskosten. Wenn sie in der Entwicklungsphase auftreten, sind sie unangenehm, aber noch kalkulierbar. Wenn sie aber nach der Auslieferung des Produktes entstehen, kann die Auswirkung schnell zu einer Gefahr für das Produkt oder das gesamte Unternehmen werden.</p>

### c) Untersuchung der Metriken

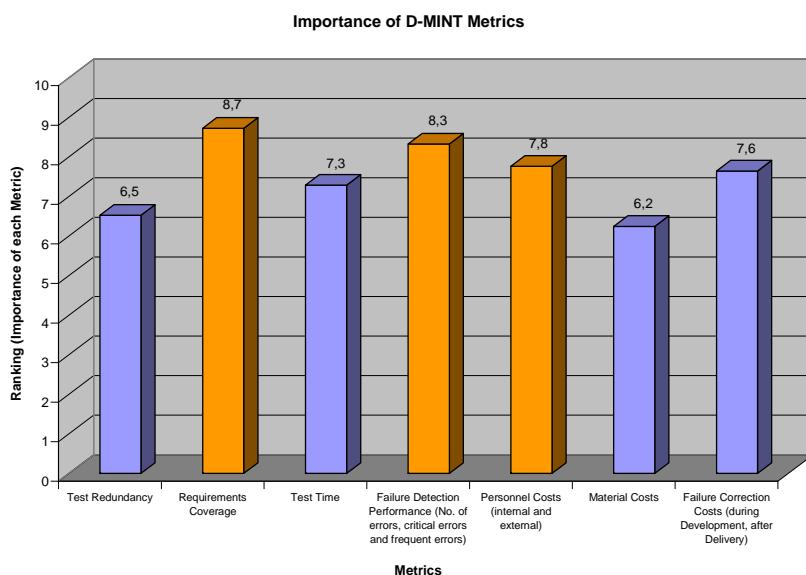
Zur Untersuchung der Metriken wurden zwei Fragestellungen beleuchtet:

- Wichtigkeit der Metriken für die Partner
- Verbesserung der Metriken durch das Projekt

Bei der ersten Fragestellung haben die Partner ihre Einschätzung für ihr individuelles Unternehmen gegeben und da es hier eine unterschiedliche Sichtweise auf Basis der unterschiedlichen Rollen im Konsortium gibt, entstand eine umfassende Abdeckung der allgemeinen Marktsituation. Das gleiche gilt für die unterschiedlichen Fallstudien, die hier aber nicht einzeln dargestellt werden.

So macht die erste Fragestellung im Hinblick auf die Softwarehersteller im Konsortium eine Aussage darüber, auf welche Aspekte der Software die potentiellen Kunden besonders achten. Dadurch können die Kundenbedürfnisse bereits früh in der Softwareentwicklung berücksichtigt werden und es kann sichergestellt werden, dass der erwartete Nutzen für die Kunden realisiert wird. Auf diese Weise kann eine starke Wettbewerbsposition erreicht werden.

Die zweite Fragestellung bezieht sich auf die tatsächlichen Verbesserungen, die aufgrund des Projektes bei den verschiedenen Metriken erzielt wurden. Auch wenn dies nur im Projekt gemessen wurde, kann doch angenommen werden, dass diese Verbesserungen auch außerhalb davon realisierbar sind, da das Konsortium mit den verschiedenen Fallstudien die jeweiligen Industrien nachbildet. So zeigen die Verbesserungspotentiale die Leistungsfähigkeit der eingesetzten Softwareprodukte aus dem Konsortium, was unter anderem als Verkaufsargument von dem jeweiligen Hersteller genutzt werden kann.



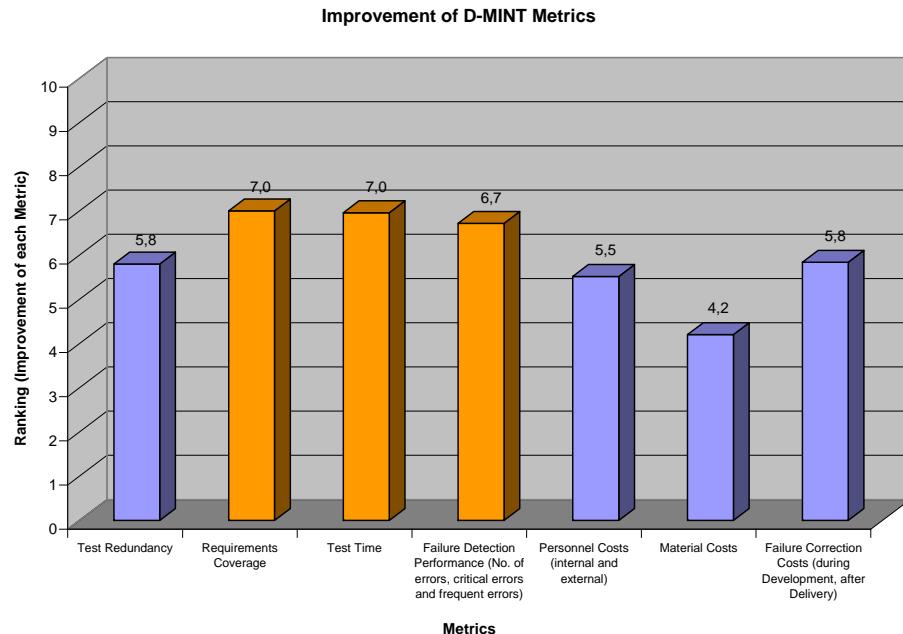
**Bild 14.10: Wichtigkeit der Metriken**

Das höchste Ranking bezüglich der Wichtigkeit der Metriken hat die Abdeckung der Anforderungen erreicht, gefolgt von der Fehlerdiagnosefähigkeit und den Personalkosten. Es ist sichtbar, dass der Fokus auf den Qualitätsaspekten liegt. Da die Anforderungen so etwas wie den formalen Auftrag des Testteams darstellen ist dieser Indikator eine Messgröße dafür, inwieweit dieser Auftrag – definiert durch den internen oder externen Kunden – erfüllt wurde.

Fehlerdiagnose bezieht sich mehr auf die Tests selbst. Die Fragestellung ist hier, inwieweit der Auftrag des Testteams erfolgreich war. Die Metrik zeigt, wie viele Fehler gefunden wurden und wie ihr Einfluss auf die Funktionen des gestesteten Systems war. Eine hohe Fehlerdiagnosefähigkeit ist zwingend erforderlich, um ein qualitativ hochwertiges Produkt zu erzeugen.

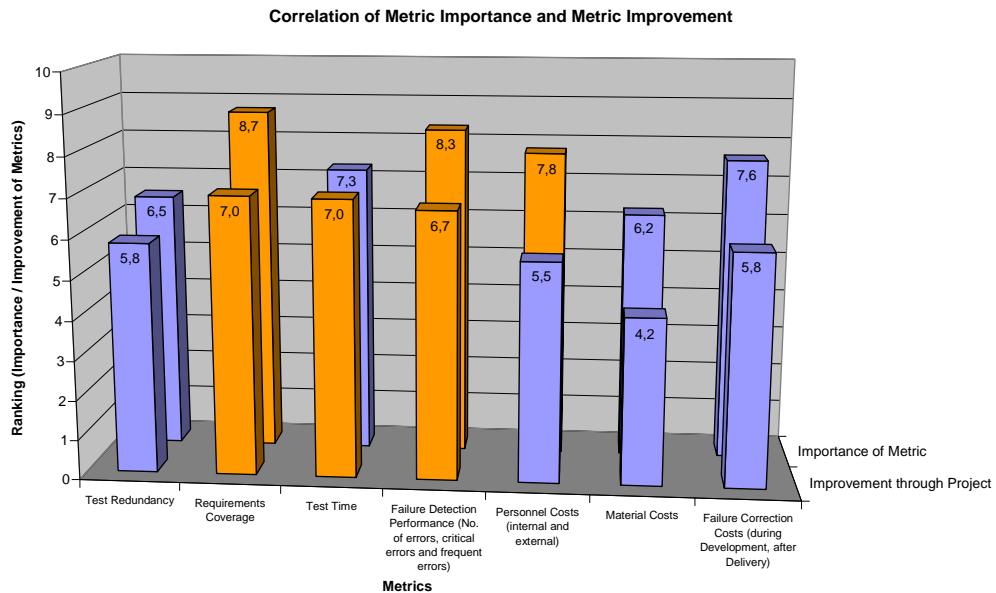
Die dritte Metrik sind die Personalkosten, die als sehr relevant eingeschätzt werden, da Testen sehr zeitaufwändige und damit kostenintensive Arbeit ist. Daher haben die Projektpartner ein hohes Interesse daran, diese Kosten unter Kontrolle zu halten.

Eine weitere Metrik aus dem Diagramm scheint wichtig zu sein. Fehlerbehebungskosten repräsentieren den monetären Aspekt der Metriken. Während die Kosten des Testprozesses wichtig sein mögen, können Kosten in der Auslieferungsphase schnell den Projekterfolg gefährden.



**Bild 14.11: Verbesserung der Metriken**

Wenn man die Verbesserungen betrachtet, die das D-MINT Projekt bezüglich der Metriken erzielt hat, stellen sich die Metriken Abdeckung der Anforderungen, Testdauer und Fehlerdiagnosefähigkeit als erfolgreichste Indikatoren heraus. Durch die Nutzung von Modellen können nicht nur die Spezifikationen strukturierter abgedeckt werden, sondern es wird auch einfacher, Testszenarien durchlaufen zu lassen, sobald sie entwickelt wurden. Dies beeinflusst die Testdauer signifikant. Wenn die Testdauer reduziert wird, kann das Testteam die zur Verfügung stehende Zeit nutzen, um weitere Fehler zu finden. Dies wird die Fehlerbehebungskosten senken, welche die vierte Metrik in diesem Ranking darstellt.



**Abbildung 14.12: Korrelation der Wichtigkeit und der Verbesserung der Metriken**

In der Korrelation der beiden untersuchten Aspekte zeigt sich der Erfolg des D-MINT Projektes, da die am wichtigsten bewerteten Metriken gleichzeitig im Wesentlichen auch die mit der größten Effizienzsteigerung sind. Außerdem haben sich auch die weniger hoch bewerteten Metriken deutlich verbessert.

Aus diesen Resultaten kann man schließen, dass modellbasiertes Testen – und speziell das D-MINT Projekt – in der Lage ist, umfassende Lösungen für eine große Zahl von Industrien bereitzustellen und so als ein Forschungs- und Entwicklungsbereich mit großem Wachstumspotential betrachtet werden kann.

## 14.2 VORAUSSICHTLICHER NUTZEN

Für die INSPIRE AG haben sich aus der Mitarbeit am D-MINT Projekt folgende Nutzenpotenziale ergeben, wobei INSPIRE den Fokus auf die ökonomischen Aspekte des Projektes gelegt hat, um die Wettbewerbsvorteile der D-MINT Methoden und Entwicklungen für potentielle Kunden darzustellen:

- Durch die Erstellung der Deliverables und die Zusammenarbeit mit den Projektpartnern konnte das Know-how im Bereich modellbasiertes Testen massiv ausgebaut werden.
- Durch die Zusammenarbeit mit den Projektpartnern konnte INSPIRE ein tieferes Verständnis für die wissenschaftlichen und technologischen Entwicklungen im MBT-Bereich erlangen, als es außerhalb des Projekts möglich gewesen wäre.
- Durch die vorgenannten Punkte konnte INSPIRE einen Wettbewerbsvorteil gegenüber anderen Beratungs- und Beteiligungsgesellschaften erzielen. So konnten bereits interessante Kontakte zu neuen potenziellen Kunden und Projektpartnern aufgebaut werden.
- Darüber hinaus wird die Zusammenarbeit mit einzelnen Partnern des Konsortiums nach dem Projektende fortgesetzt werden.

## 14.3 FORTSCHRITT BEI ANDEREN STELLEN

keine

## 14.4 ERFOLGTE ODER GEPLANTE VERÖFFENTLICHUNGEN

keine

## 15. LISTE DER VERÖFFENTLICHUNGEN

Dieser Abschnitt fasst die Veröffentlichungen der deutschen Projektpartner von D-MINT zusammen:

- [D-MINT 1] I. Schieferdecker , J. Großmann: Testing Embedded Control Systems with TTCN-3, An Overview on TTCN-3 Continuous, 5th IFIP Workshop on Software Technologies for Future Embedded and Ubiquitous Systems, Santorini, Greece, April 07.
- [D-MINT 2] I. Schieferdecker: Model-Based Testing, ASQF Seminar, Erlangen, Mai 07.
- [D-MINT 3] J. Grabowski, H. Neukirchen, I. Schieferdecker, D. Vega, B. Zeiss: An ISO 9126-based Quality Model to Assess the Quality of TTCN-3 Test Specifications. 4th TTCN-3 User Conference, Stockholm (S), Mai 07.
- [D-MINT 4] R. Eschbach, Sequence Based Specification and Statistical Testing, Siemens TS Software Initiative Workshop, Braunschweig, Germany, May 24, 2007
- [D-MINT 5] C. Robinson-Mallett, Model-based Testing, Siemens TS Software Initiative, Braunschweig , Germany, May 24, 2007
- [D-MINT 6] T. Bauer, T. Beletski, F. Böhr, R. Eschbach, D. Landmann, J. H. Poore, From Requirements to automated statistical testing, 4th International ICSE workshop on Software Engineering for Automotive Systems, SEAS 2007, Minneapolis, USA, May 26, 2007
- [D-MINT 7] A. Vouffo-Feudjio, I. Schieferdecker: Pattern-based development of TTCN-3 test suites. 4th TTCN-3 User Conference, Stockholm (S), Mai 07.
- [D-MINT 8] D. Vega, I. Schieferdecker, G. Din: Test Data Variance as a Test Quality Measure: Exemplified for TTCN-3, Testing of Software and Communicating Systems, 19th IFIP TC6/WG6.1 International Conference, TestCom 2007, 7th International Workshop, FATES 2007, Tallinn, Estonia June 07.
- [D-MINT 9] J. Zander-Nowicka, A. Marrero Pérez, I. Schieferdecker: From Functional Requirements through Test Evaluation Design to Automatic Test Data Retrieval - a Concept for Testing of Software Dedicated for Hybrid Embedded Systems, The 2007 World Congress in Computer Science, Computer Engineering, & Applied Computing; The 2007 International Conference on Software Engineering Research and Practice, Las Vegas (USA), Juni 07.
- [D-MINT 10] I. Schieferdecker: Modellbasiertes Testen, OBJEKTspektrum, Heft 03/2007, SIGS-DATACOM, Juni 2007
- [D-MINT 11] J. Großmann , I. Schieferdecker: Testing Embedded Control Systems with TTCN-3, Enhancing TTCN-3 with Concepts for Continuous Streams. STTT Special Issue on TTCN-3 Evolution, Juni 07.
- [D-MINT 12] I. Schieferdecker: Test Generation as Model Transformation, TAROT Summer School, June 07.
- [D-MINT 13] J. Zander-Nowicka, A. Marrero Pérez, I. Schieferdecker, Z. R. Dai: Test Design Patterns for Embedded Systems, 10th International Conference on Quality Engineering in Software Technology, CONQUEST 2007, Potsdam (D) Sept. 2007.
- [D-MINT 14] J. Zander-Nowicka: Model Based Testing of Embedded Real-Time Systems, University of California in San Diego (U.S.A.), Computer Science and Engineering Department, Service-Oriented Software & Systems Engineering Laboratory, Sep. 07.
- [D-MINT 15] I. Schieferdecker: Test und Analyse sicherheitsgerichteter Systeme. Deutscher Kongress, Stuttgart, Oct 07.
- [D-MINT 16] I. Schieferdecker: Test Specifications as a Basis for common Test Requirements between OEM and Vendor. Systematic Testing Conference, Düsseldorf, Oct. 07.
- [D-MINT 17] T. Bauer, Model-based statistical testing of technical software systems, Systematic Testing Conference, Düsseldorf, Germany, October 22, 2007
- [D-MINT 18] C. Robinson-Mallett, R.-M. Hierons, J. H. Poore, T. Bauer, Using partial models to support the testing of distributed systems, The 11th IASTED International Conference on, Cambridge-MA, USA, November 19-21, 2007

- [D-MINT 19] I. Schieferdecker: Integrierte Systementwicklung und Test – Modelle als Grundlage effizienter Qualitätssicherung, MID Insight'07, Nov. 07
- [D-MINT 20] I. Schieferdecker: A test model quality framework. Model-Based Engineering of Embedded Real-Time Systems, Dagstuhl (D), Nov. 07.
- [D-MINT 21] J. Zander-Nowicka: Reactive Testing and Test Control of Hybrid Embedded Software. 5th Workshop on System Testing and Validation STV07, in conjunction with ICSSEA 2007. Paris (F) December 07
- [D-MINT 22] Thomas Deiß, Model-based testing at Nokia Siemens Networks, Seminar Universität Göttingen, 2007
- [D-MINT 23] Jens Lüdemann, Eckard Bringmann, Andreas Krämer, Messeausstellung, Embedded World 2008, Nürnberg, Germany, 2008
- [D-MINT 24] Eckard Bringmann, Automated model-based testing of control software with TPT, Embedded World Conference 2008, Nürnberg, Germany 2008
- [D-MINT 25] I. Schieferdecker: Paradigmenwechsel im Softwaretest, Fraunhofer IESE Akademie, Kaiserslautern, Feb. 08
- [D-MINT 26] iXtronics Präsentation auf der Embedded World 2008 in Nürnberg, Feb. 2008.
- [D-MINT 27] Stefan Lachmann, Andreas Krämer, Systematischer Test des kontinuierlichen Verhaltens automobiler Softwaresysteme, 3. Tagung: Simulation und Test für die Automobilelektronik, Berlin, Germany 2008
- [D-MINT 28] A. Hoffmann, I. Schieferdecker: Modellgestütztes Testen komplexer Systeme. 5tes GI/ASQF/IRC, Schloß-Steinhöfel-Seminar, „Zukunft gestalten“, Steinhöfel (D), Apr. 08.
- [D-MINT 29] Eckard Bringmann, Andreas Krämer, Model-Based Testing of Automotive Systems, First International Conference on Software Testing, Verification, and Validation, ICST 2008, Lillehammer, Norway, 2008
- [D-MINT 30] Thomas Deiß, Trends in TTCN-3: Language and Usage, Software & Systems Quality Conference, Düsseldorf, 2008, [http://www.sqs-conferences.com/de/2008/program\\_08.htm](http://www.sqs-conferences.com/de/2008/program_08.htm)
- [D-MINT 31] Jens Lüdemann, Eckard Bringmann, Andreas Krämer, Automotive Testing Expo 2008, Stuttgart, Germany 2008
- [D-MINT 32] K.-D. Engel, A. Rennoch, I. Schieferdecker: Architecture-driven Test Development, MoTiP 08, Berlin, June 08.
- [D-MINT 33] T. Bauer, F. Böhr, R. Eschbach, On MiL, HiL, Statistical Testing, Reuse, and Efforts, 1st Workshop on Model-based Testing in Practice - MoTiP 2008, Berlin, Germany, 12-Jun-08, Fraunhofer IRB
- [D-MINT 34] Detlef Streitferdt, Philipp Nenninger, Carlos Bilich, Florian Kantz, Thomas Bauer, Robert Eschbach, "Model-based Testing in the Automation Domain, Safety Enabled", in Proceedings of the 1st Workshop on Model-based Testing in Practice, MoTiP at the Fourth European Conference on Model Driven ArchitectureFoundations and Applications (ECDMA), Berlin, 2008.
- [D-MINT 35] Florian Kantz, Detlef Streitferdt, and Philipp Nenninger, "Generic Interface for Modelbased Testing in theAutomation Domain", in Proceedings of the 1st Workshop on Model-based Testing in Practice, MoTiP at the ECDMA Conference, Berlin, 2008.
- [D-MINT 36] Detlef Streitferdt, Philipp Nenninger, "Requirements on Testing in the Development of Embedded Automation Systems", in Proceedings of VDI Automation Kongress, Baden-Baden, 2008.
- [D-MINT 37] Detlef Streitferdt, Philipp Nenninger, Holger Kaul, Florian Kantz, "Low Power Development & Testing", in Proceedings of the Workshop on Low Power Design Impact on Test and Reliability at ETS08, 2008.
- [D-MINT 38] A. Vouffo Feudjio: A Unified Approach to Test Modelling, MoTiP 08, Berlin, June 08.
- [D-MINT 39] T. Bauer, H. Eichler, A. Rennoch, 1st Workshop on model-based testing in practice (MoTiP'08) at the 4th European Conference on Model-Driven Architecture Foundations and Applications (ECMDA 2008), Berlin, Germany, 12 June 2008, MoTiP'08 Workshop Proceedings, Publisher: Fraunhofer IRB Verlag, ISBN 978-3-8167-7624-6, 12 June 2008

- [D-MINT 40] J. Zander-Nowicka: Model-Based Testing of Real-Time Embedded Systems for Automotive Domain, Symposium on Quality of Embedded Systems, Berlin, Juni 08.
- [D-MINT 41] I. Schieferdecker: Testing Embedded systems, TAROT summer school, Bath, UK, Juni 08.
- [D-MINT 42] C. Hein, T. Ritter, M. Wagner: Model-Driven Tool Integration Framework – ModelBus. 4th European Conference on Model Driven Architecture Fondations and Applications (ECMDA 2008), Berlin, Juni 08.
- [D-MINT 43] Zander-Nowicka, Justyna; Schieferdecker, Ina: Systematic Test Data Generation for Embedded Software. SERP'08, The 2008 International Conference on Software Engineering Research and Practice, Las Vegas (USA), Jul 08.
- [D-MINT 44] Vega, Diana-Elena; Din, George; Schieferdecker, Ina: TTCN-3 Test Data Analyser using Constraint Programming. Nineteenth international Conference on Systems Engineering (ICSENG 2008), Las Vegas (USA), Aug 08.
- [D-MINT 45] Theofanis Vassiliou-Gioles, Model-based testing with U2TP and TTCN-3, 4th European Conference on Model Driven Architecture Fondations and Applications, Berlin, Germany, 2008
- [D-MINT 46] Bogdan Stanca-Kaposta, Stephan Pietsch, Model-based testing with UTP and TTCN-3 and its Application to HL7, Conference on Quality Engineering in Software Technology, Potsdam, Germany, 2008
- [D-MINT 47] Zander-Nowicka, Justyna; Rennoch, Axel; Bauer, T.; Eschbach, Dr. Robert: Modellbasierte Testentwicklung in der industriellen Anwendung, Objectspectrum "Testing", SIGS Datacom, Oktober 08.
- [D-MINT 48] A. Vouffo: Quality of Test Specification by Application of Patterns. 2nd Workshop on Software Patterns and Quality (SPAQu'08), Nashville, Tennessee, USA, Oct. 08.
- [D-MINT 49] Vega, Diana-Elena; Din, George; Schieferdecker, Ina: Automated Maintainability of TTCN-3 Test Suites based on Guideline Checking: The 6th IFIP Workshop on Software Technologies for Future Embedded and Ubiquitous Systems (SEUS 2008), Capri (I), Oct 08.
- [D-MINT 50] Vega, Diana-Elena; Din, George; Schieferdecker, Ina; Arbanowski, , Stefan: Application of Clustering Methods for Analysing of TTCN-3 Test Data Quality, The Third International Conference on Software Engineering Advances ICSEA 2008, Sliema (Malta), Oct. 08
- [D-MINT 51] Präsentation auf der SPS IPC DRIVES 2008 in Nürnberg. Vorgestellt wurde die Toolkopplung zwischen der iXtronics Software CAMeL-View und der PikeTec Software TPT am Beispiel des ABB Softstarters. Nov. 2008.
- [D-MINT 52] T. Bauer, A. Rennoch, I. Schieferdecker: Model-Based Testing and its use in the ITEA2 project D-MINT, ICSSEA 2008, 21st edition of the ICSSEA International Conference on Software & Systems Engineering and their Applications, Paris (F) Dez 08.
- [D-MINT 53] Vouffo, A. ,Kingni, A.: Model-Driven Engineering of IMS Tests. 6th Workshop on System testing and validation (STV08), Madrid (E), Dezember 08.
- [D-MINT 54] Krämer, Gut, Baderschneider, Systematischer Test von Fahrerassistenz-Systemen in einer virtuellen Verkehrsumgebung, Hardware-in-the-Loop Simulation, Kassel, Germany 2008
- [D-MINT 55] C. Robinson-Mallett, R.-M. Hierons, J. H. Poore, P. Liggesmeyer, Using Communication Coverage Criteria and Partial Model Generation to assist Software Integration Testing , Software Quality Journal, Vol.16, 2008
- [D-MINT 56] J. Zander-Nowicka: Model-based Testing of Embedded Systems in the Automotive Domain, TU Berlin, IRB Verlag, Januar 2009
- [D-MINT 57] Jens Lüdemann, Eckard Bringmann, Andreas Krämer, Messeausstellung, Embedded World 2009 Nürnberg, Germany 2009
- [D-MINT 58] S. Weiß, Model based Control Testing using TPT, Webex Meeting, D-Mint Webinar, Berlin, Germany 2009
- [D-MINT 59] J. Zander-Nowicka: Quality assurance of automotive hybrid embedded software. 5th IFIP Summer School & 2nd ACM/IEEE ICSE Workshop, Cape Town, SouthAfrica, March 09.
- [D-MINT 60] Bogdan Stanca-Kaposta, Executable Test-Models – From UML Test Profile to Test Report, ASQF Fachgruppe Software-Test Berlin-Brandenburg, Berlin, Germany, 2009

- [D-MINT 61] Präsentation auf dem 6. Paderborner Workshop „Entwurf mechatronischer Systeme“. 2-3.4.2009, Heinz-Nixdorf Forum Paderborn, Begleitende Fachausstellung.
- [D-MINT 62] Theofanis Vassilou-Gioles, Testautomatisierung mit TTCN-3 am Beispiel des UML Testprofils, Software & Systems Quality Conference (SQC), Düsseldorf, Germany, 2009
- [D-MINT 63] Alexander Lorenz, Derivation and Automatic Generation of TTCN-3 Test Cases from UML State Machine Test Models, Master Thesis, FernUniversität Hagen, Germany, 2008
- [D-MINT 64] Alain Vouffo: Model-driven Functional Test Engineering for Service Centric Systems, TridentCom09, Washington DC, April 09.
- [D-MINT 65] F. Zimmermann, R. Eschbach, J. Kloos, T. Bauer, Risk-based Statistical Testing: A refinement-based approach to the reliability analysis of safety-critical systems, 2th European Workshop on Dependable Computing Toulouse, France, 14-15 May 2009
- [D-MINT 66] Jens Lüdemann, Automatic ASAM MCD-3 supported model based testing, Technology Forum at the Automotive Testing Expo 2009, Stuttgart, Germany 2009
- [D-MINT 67] Jens Lüdemann, Eckard Bringmann, Andreas Krämer, Automotive Testing Expo 2009, Stuttgart, Germany 2009
- [D-MINT 68] Florian Kantz, Thomas Ruschival, Philipp Nenninger, Detlef Streitferdt, "Testing with Large Parameter Sets for the Development of Embedded Systems in the Automation Domain", in Proceedings of the 2nd International Workshop on Component-Based Design of Resource-Constrained Systems (CORCS 2009) at the 33rd Annual IEEE International Computers, Software and Applications Conference, Seattle, USA, 2009.
- [D-MINT 69] J. Kloos, .R. Eschbach, Generating System Models for a Highly Configurable Train Control System Using A Domain-Specific Language: A Case Study, Workshop paper, 5th Workshop on Advances in Model Based Testing (A-MOST 2009) at the 2nd International Conference on Software Testing, Verification, and Validation (ICST'09), Denver, Colorado, USA, 2009
- [D-MINT 70] Thomas Ruschival, Philipp Nenninger, Florian Kantz, Detlef Streitferdt, "Test Case Mutation in Hybrid State Space for Reduction of No-Fault-Found Test Results in the Industrial Automation Domain", in Proceedings of the 2nd International Workshop on Industrial Experience in Embedded Systems Design (IEESD 2009) at the 33rd Annual IEEE International Computers, Software and Applications Conference, Seattle, USA, 2009.
- [D-MINT 71] Offener Workshop CAMeL-View Essentials in Paderborn; Vermittlung von Techniken zum modellbasierten Testen, Juni 2009.
- [D-MINT 72] T. Bauer, A. Rennoch: TTCN-3 and its role and usage in MBT from the D-MINT perspective, TTCN-3 user conference, Sophia Antipolis (F), June 09.
- [D-MINT 73] G. Din, K.-D. Engel: An Approach for Test Derivation from System Architecture Models applied to Embedded Systems, MoTiP workshop, Enschede (NL), June 09.
- [D-MINT 74] T. Bauer, R. Eschbach, M. Größl, T. Hussain, D. Streitferdt and F. Kantz, Combining Combinatorial and Model-based Test Approaches for Highly Configurable Safety-critical Systems, 2nd Workshop on Model-based Testing in Practice (MoTiP 2009), Enschede, The Netherlands, 23 June 2009
- [D-MINT 75] T. Bauer, H. Eichler, A. Rennoch, S. Wieczorek, 2nd Workshop on model-based testing in practice (MoTiP'09) at the 5th European Conference on Model-Driven Architecture Foundations and Applications (ECMDA 2009), Enschede, The Netherlands, 23 June 2009; MoTiP'09 Workshop Proceedings, CTIT Workshop Proceedings Series WP09-07, ISSN 0929-0671, Enschede, The Netherlands, 23 June 2009
- [D-MINT 76] A. Vouffo, I. Schieferdecker: A Pattern Language of Test Modelling for Reactive Software Systems. Symposium at European conference on patterns (EuroPLoP2009), July 09.
- [D-MINT 77] Colin Willock, D-MINT: Bringing Model-Based Testing to Industry, invited talk, 8th Intl. Conference on Software QA and testing for Embedded Systems, 2009, Bilbao
- [D-MINT 78] A. Vouffo: Model-driven Functional Test Engineering for Service Centric Systems, Workshop on Visual formalisms for patterns, Oregon (USA) 24.9.2009.

- [D-MINT 79] A. Hoffmann, A. Rennoch, I. Schieferdecker, N. Radziwil: A Generic Approach for Modeling Test Case Priorities with Applications for Test Development and Execution. Modellbasiertes Testen (MoTes 2009), Lübeck (D) 29.09.2009.
- [D-MINT 80] F. Zimmermann, R. Eschbach, J. Kloos, T. Bauer. Risikobasiertes statistisches Testen (German article), Workshop paper, 28. TAV-Treffen, Dortmund, Germany, 2009
- [D-MINT 81] G. Din, K. Engel, A. Rennoch: Architekturbasierte Testentwicklung für eingebettete Systeme, OBJEKTspektrum, Ausgabe Testing/2009, SIGS Datacom, Oktober 09.
- [D-MINT 82] Robert Eschbach, Risikobasiertes statistisches Testen sicherheitskritischer eingebetteter Systeme, 7th SafeTRANS Industrial Day, Friedrichshafen Germany, 19.11.09
- [D-MINT 83] D-MINT project consortium: Common Approach to Architecture-Driven Testing, D-MINT whitepaper. Nov. 2009. [http://www.d-mint.org/public/CommonApproach\\_WhitePaper\\_DMINT.pdf](http://www.d-mint.org/public/CommonApproach_WhitePaper_DMINT.pdf)
- [D-MINT 84] Dr. Jens Lüdemann, Systematischer Softwaretest sicherheitskritischer Systeme Technology Forum DESIGN&ELEKTRONIK-Entwicklerforum »Embedded goes medical«, Leipzig, Germany 2009
- [D-MINT 85] Dr. Stefan Lachmann, Embedded Software Systeme systematisch und durchgängig Testen, DESIGN&ELEKTRONIK-Entwicklerforum »Embedded-System-Entwicklung« Ludwigsburg, Germany 2009
- [D-MINT 86] Dr. Eckard Bringmann, Testen von eingebetteten Steuerungs- und Regelungssystemen ASQF-Fachgruppen-Treffen Test, Berlin, Germany 2009
- [D-MINT 87] A. Vouffo: Testmodellierungen in Industrie und Forschung - Überblick und Beispiele. ASQF Fachreferat. Berlin, Mai 2010.
- [D-MINT 88] T. Bauer, H. Eichler, M.-F. Wendland, S. Wieczorek, 3rd Workshop on model-based testing in practice (MoTiP'10) at the 6th European Conference on Model-Driven Architecture Foundations and Applications (ECMFA 2010), Paris, France, 16 June 2010
- [D-MINT 89] J. Zander, I. Schieferdecker: Model-based Testing of Embedded Systems Exemplified for the Automotive Domain, Book chapter in In: Behavioral Modeling for Embedded Systems and Technologies: Applications for Design and Implementation, Editors: Gomes, L., Fernandes, J. M., Idea Group Inc (IGI) July 2010.
- [D-MINT 90] Ch. Buckl, A. Knoll, I. Schieferdecker, J. Zander: Model-Based Analysis and Development of Dependable Systems. Book chapter In: Model-Based Engineering of Embedded Real-Time Systems, LNCS State of the Art-Surveys for the MBEERTS Dagstuhl Seminar, 2010.
- [D-MINT 91] A. Rennoch, S. Lüdtke, K.D. Engel: Architecture-Driven Testing using TTCN-3. Submitted for ICSSEA'2010, Paris, Dec. 2010.

Die folgende Liste enthält die wesentlichen Beiträge der deutschen Projektpartner zu den finalen internationalen D-MINT Projektberichten. Öffentliche Versionen der Arbeitspakete 2 und 3 sind verfügbar unter <http://www.d-mint.org/assetbox.html>.

- [D-MINT 92] D.1.1.2.v10.Final.NSN\_DemonstratorDescription, Milestone 2
- [D-MINT 93] D.1.1.D1\_2\_Daimler.v1.1.Final.Daimler\_demonstrator, Milestone 2
- [D-MINT 94] D.1.2.v2.0.FINAL.CaseStudyRequirements\_IndAutomation, ABB Fallstudie, Milestone 2
- [D-MINT 95] D.1.1.D1\_3.v1.0.Final.ABB\_Demonstrator\_Manual , Milestone 3
- [D-MINT 96] D.1.1.D1\_3.v1.0.Final.Daimler\_Demonstrator\_Manual, Milestone 3
- [D-MINT 97] D.1.1.D1\_3.v1.0.Final.NSN\_Demonstrator\_Manual, Milestone 3
- [D-MINT 98] D.1.3.D1\_5.v1.0.Final.ABB\_Case\_Study\_Evaluation, Milestone 5
- [D-MINT 99] D.1.3.D1\_5.v1.0.Final.Daimler\_Case\_Study\_Evaluation. Milestone 5
- [D-MINT 100] D.1.3.D1\_5.v1.0.Final.NSN\_Case\_Study\_Evaluation, Milestone 5
- [D-MINT 101] D.2.1.v2.1\_FINAL.TestModellingTestGeneration, Methoden des Model-basierten Testen in den behandelten Domänen, Milestone 5

- [D-MINT 102] D.2.6.v2.0\_FINAL.TestProcessTestValidation, Ansätze zur Testauswahl, Testqualität, Validierung sowie Bewertung der Fallstudien, Milestone 5
- [D-MINT 103] D.3.1.v1.0 Test Generation and Refinement, Testgenerierung mit UML und TPT, Milestone 2
- [D-MINT 104] D.3.2.v1.0.FINAL.TestManagement, Verteilte Testprozeßsteuerung, Milestone 3
- [D-MINT 105] D.3.3.v1.0.FINAL.TestQuality, Konzepte und Implementierungen in D-MINT, Milestone 4
- [D-MINT 106] D.3.4.v1.0.FINAL.IntegratedDescriptionOfAllToolChainComponents.pdf, Werkzeugbeschreibungen dt. Partner, Milestone 5
- [D-MINT 107] D.4.1.1.v3.1.Final.INSPIRE.ExploitationPlanRevision3.pdf , Milestone 5
- [D-MINT 108] D.4.2.2.v1.0.Final.INSPIRE.ProductisableSolutionSet.pdf , Milestone 5
- [D-MINT 109] D4.3.1.v2.0.Final.INSPIRE.IndustrialApplicationData.pdf, Milestone 5
- [D-MINT 110] D.5.2.1.v20.FINAL.Coaching&Training.pdf, Milestone 5
- [D-MINT 111] D.5.3.1.v10.FINAL.StandardizationReport .pdf, Milestone 5

## 16. VERÖFFENTLICHUNGEN ANDERER STELLEN/AUSSERHALB DES PROJEKTS

Die folgende Liste enthält weitere Veröffentlichungen, die nicht von den Projektpartnern erstellt wurden, auf die im Text jedoch verwiesen wird:

- [A1] Dustin E., Rashka J., Paul J., Automated Software Testing: Introduction, Management, and Performance, AddisonWesley, 1999
- [A2] Pang C., Biscotti F., LatimerLivingstone N. S., Forecast: Application Development Software, EMEA, 20022008, Gartner Group, 2005
- [A3] Poston R. M., Automating SpecificationBased Software Testing, IEEE Computer Society, 1996
- [A4] B. Beizer, Software Testing Techniques, Van Nostrand Reinhold, New York,1989
- [A5] Douglass B. P., RealTime UML: Developing Efficient Objects for Embedded Systems, , Addison-Wesley, 1998
- [A6] Selic B., Gullekson G., Ward P. T., RealTime ObjectOriented Modeling, Wiley, 1994
- [A7] Prowell S. J., Trammell C. J., Linger R., Poore J. H., Cleanroom Software Engineering, Addison-Wesley, 1999
- [A8] Yourdon E., Constantine L.L., Structured Design, Yourdon Press, 1975
- [A9] DeMarco T., Structured Analysis and System Specification, Prentice Hall, 1978
- [A10] Ward P., Mellor S., Structured Development for RealTime Systems, Yourdon Press, 1985
- [A11] D'Souza D. F., Wills A. C.,Components and Frameworks with UML: The Catalysis Approach, AddisonWesley, 1998
- [A12] Kruchten, P. B., The Rational Unified Process. An Introduction, AddisonWesley, 2000
- [A13] Atkinson, C., Bayer, J., Bunse, C. et al. ComponentBased Product Line Engineering with UML. AddisonWesley, 2001
- [A14] ETSI ES 201-873-1: The Testing and Test Control Notation TTCN-3: Core Language.
- [A15] UTP: UML Testing Profile, OMG. www.omg.org.
- [A16] Liggesmeyer P., Software Qualität, 2002, Spektrum Akademischer Verlag
- [A17] Clarke E.M., Grumberg, O., Peled, A., Model Checking, MIT Press, 1999
- [A18] ITU, Specification and Description Language (SDL), Z.100, 2002, www.itu.int
- [A19] Unified Modeling Language 2.0 Reference Manual, Object Management Group, www.omg.org
- [A20] Open SystemC Initiative (OSCI). SystemC 2.0.1 Language Reference Manual, 2003,
- [A21] SysML Partners, System Modeling Language (SysML) specification, v1.0, 2005, www.sysml.org

 <b>D-MINT</b> Deployment of Model-Based Technologies to Industrial Testing	<b>D-MINT Abschlussbericht</b>	Seite : 80 of 81 <hr/> Version: 1.0 Datum: 30.06.2010 <hr/> Status : Final
---	--------------------------------	---

- [A22] Guide to Model DrivenArchitecture, Object Management Group, [www.omg.org](http://www.omg.org)
- [A23] ETAS, ASCET SD: [en.etasgroup.com/products/ascet/](http://en.etasgroup.com/products/ascet/)
- [A24] iLogix, Rhapsody: [www.ilogix.com](http://www.ilogix.com)
- [A25] IBM, Rational Rose Technical Developer: [www306.ibm.com/software/awdtools/developer/technical/](http://www306.ibm.com/software/awdtools/developer/technical/)
- [A26] Telelogic, Telelogic TAU: <http://www.telelogic.com>
- [A27] Modelica, [www.modelica.org](http://www.modelica.org)
- [A28] The Mathworks, MatLab Simulink: [www.mathworks.com](http://www.mathworks.com)
- [A29] National Instruments, Labview: [www.labview.com](http://www.labview.com)
- [A30] CAMELView TestRig, [www.ixtronics.com](http://www.ixtronics.com)
- [A31] MATELO, Markov Test Logic Automated Statistical Testing Suite for Software Validation, [www.alitec.net/matelo](http://www.alitec.net/matelo)
- [A32] PETS, Prediction of Software Error Rates based on Test and Software Maturity Results, [www.ist-pets.com](http://www.ist-pets.com)
- [A33] MAPP, Multivariate Approach for statistical process control and cleaner production, [www.mapp.nu](http://www.mapp.nu)
- [A34] ranTEST, Entwicklung einer praxistauglichen Technik für das risikominimierende, anforderungsbasierte Testen von SoftwareSystemen , [www.rantest.de](http://www.rantest.de)
- [A35] T. Ringler et al.: An Approach to Tool-based Design of Electrics/Electronics Architectures, ATZelektronik worldwide Edition: 2008-01.

## 17. ABKÜRZUNGSVERZEICHNIS

AP	Arbeitspaket
BSC	Base Station Controller
CORBA	Common Object Request Broker Architecture
CTE	Classification Tree Editor
CTMF	Conformance Testing Methodology and Framework
D-MINT	Deployment of Model-Based Technologies
ECU	Electronic Control Unit
ETSI	European Telecommunication Standardization Institute
FPP	Full Project Proposal
FMCT	Formal Methods in Conformance Testing
GSM	Global System for Mobile Communications
HTTP	Hypertext Transfer Protocol
IDL	Interface Definition Language
IPv4	Internet Protocol version 4
IMS	IP Multimedia Subsystem
ISO	International Standardisation Organization
ITEA	Information Technology for European Advancement
ITU	International Telecommunications Union

 <b>D-MINT</b> <small>Deployment of Model-Based Technologies to Industrial Testing</small>	<b>D-MINT Abschlussbericht</b>	Seite : 81 of 81
		Version: 1.0 Datum: 30.06.2010
		Status : Final

MBT	Modell-based Testing
MoTiP	Model-based Testing in Practice
MSC	Message Sequence Chart
NE	Netzwerkelement
PDE	Positioning Determination Entity
RMI	Remote Method Invocation
RPP	Remote Positioning Protocol
SA	SUT Adapter
SMLC	Serving Mobile Location Center
SUT	System Under Test
TCI	TTCN-3 Control Interface
TLI	TTCN-3 Logging Interface
TPT	Timed Partition Testing
TRI	TTCN-3 Runtime Interface
TTCN-3	Testing and Test Control Notation version 3
UML	Unified Modelling Language
U2TP	UML2 Testing Profile
XML	eXtensible Markup Language
ZE	Zuwendungsempfänger