

Safety first

Sicherheitskonzepte für Rechnersysteme



Abbildung 1:
Sichere Systeme sollen
Unfälle vermeiden

Vorsicht ist die Mutter der Porzellankeise. Passen Sie auf! Ein technischer Fehler könnte Ihren elektronischen Kalender ausfallen lassen. Ihr Adressbuch auch. Und dann kann es peinlich werden: Man blamiert sich bis auf die Knochen, wenn man nicht mehr weiß, mit wem man sich wann verabredet hat. Also speichert man die Kalender- und Adressdaten regelmäßig auf einem PC, besser noch auf mehreren, so dass man im Fehlerfall dort nachsehen kann. Fehler sind heimtückische Gegner von Informatik-Systemen.

Geeignete Gegenmaßnahmen machen ein Rechen-system widerstandsfähig, so dass trotz Fehlern ein hohes Maß an *Safety* und trotz Angriffen ein hohes Maß an *Security* gewährleistet werden kann. Im Deutschen werden *Safety* und *Security* unter dem Begriff *Sicherheit* zusammengefasst.

Wenn es um den Betrieb von großen, unternehmenskritischen Rechnernetzen oder gar um die Automatisierung von Techniksystemen geht, die Menschen gefährden können, spielt die Sicherheit eine zentrale Rolle. Im Bahn- und Luftverkehr sind zahlreiche Rechen-systeme redundant ausgelegt, um zu vermeiden, dass technische Fehler zu Unfällen führen. Auch im Automobil werden zunehmend sicherheitskritische Aufgaben von Rechnern übernommen, die hohen Zuverlässigkeitsanforderungen gerecht werden müssen.

täglichen Umgang mit einem System bekommt man kaum ein „Gefühl“ für Sicherheit, weil ernste Fehler nur höchst selten auftreten. Für Experten, die sichere Systeme konstruieren, ergeben sich daraus zwei Herausforderungen: Zum einen müssen die Sicherheitskonzepte so effizient wie nur irgend möglich realisiert werden. Zum anderen muss ihre Wirksamkeit formal nachgewiesen werden – rigoros gegenüber allen Fehlern, auch denen, die man nach menschlichem Ermessen als untypisch einstuft. Die Forschungsgruppe „Verlässlichkeit von Rechen-systemen“ des Essener Instituts für Informatik und Wirtschaftsinformatik entwickelt unter Leitung von Klaus Echtele *Safety*-Methoden und Fehlertoleranzverfahren der nächsten Generation.

Höchstmögliche Verlässlichkeit

Wenn wir einem Rechen-system besonders wichtige Aufgaben übertragen, darf es gegenüber Fehlern nicht anfällig sein. Rechner sollen nicht nur keine Schäden verursachen – sie sollen sogar gefährliche Zustände vermeiden, in denen Schäden eintreten könnten. Stattdessen fordern wir höchstmögliche *Verlässlichkeit* (engl. dependability), ein umfassender Oberbegriff, der Sicherheit im Sinne von *Safety* und *Security*, aber auch *Zuverlässigkeit* und *Verfügbarkeit* einschließt.

Die Verlässlichkeit von Systemen wird durch Redundanz verbessert. Ausgeklügelte Verfahren der *Fehlertoleranz* nutzen diese, so dass ein Rechen-system auch dann noch funktioniert, wenn einzelne Teile ausgefallen sind. Beispielsweise kann ein Mehrrechnersystem aus drei Prozessoren Fehlfunktionen eines einzelnen Prozessors ver-

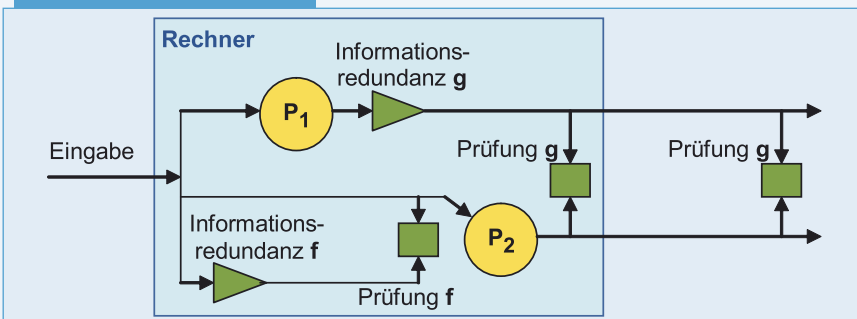


Abbildung 2:
Virtuelles Duplexsystem mit
den systematisch diversitären
Prozessen P1 und P2

Hochgradige Sicherheit ist oft nur durch teure Maßnahmen zu erreichen, die zusätzliche Hardware erfordern und beträchtliche Rechenzeit kosten, aber auch zu Nutzungseinschränkungen führen können. Und selbst dann ist es schwierig, die erreichte Sicherheit zu überprüfen. Im

kraften, wenn als Fehlertoleranzverfahren eine 2-von-3-Mehrheitsentscheidung eingesetzt wird. Schon dieses einfache Beispiel wirft ein schwieriges Problem auf: Wie sind Fehler des Mehrheitsentscheiders (engl. voter) zu tolerieren? Ganz gleich, wo der Fehler liegt: Das System muss sich sicher verhalten. Je nach Anwendung muss im Fehlerfall die volle Funktionsfähigkeit erhalten bleiben (*fail-operational*, typisch in der Luftfahrt); in anderen Bereichen genügt das Anhalten des Systems (*fail-safe*, typisch im Bahnverkehr).

Die zur Fehlertoleranz erforderlichen Mittel können die Kosten stark in die Höhe treiben. Zum einen muss besonders zuverlässige Hardware eingesetzt werden. Zum anderen sind zahlreiche Komponenten redundant auszulegen, beispielsweise dreifach. Darüber hinaus handelt es sich bei fehlertoleranten Rechnern oft um Spezialgeräte, die nicht mit Standard-Betriebssystemen und -Software arbeiten. Sie benötigen speziell entwickelte Software, was die Entwicklungskosten stark erhöht. Außerdem müssen sicherheitskritische Systeme inklusive der darin enthaltenen Rechner oft zertifiziert werden, was ebenfalls teuer ist.

Kombinierte Redundanz

Am Lehrstuhl „Verlässlichkeit von Rechensystemen“ wurden zahlreiche Fehlertoleranzverfahren entwickelt, die ein günstiges Verhältnis von Zuverlässigkeit und Redundanzaufwand erzielen. Sie kombinieren verschiedene Redundanzformen in besonderer Weise miteinander: beispielsweise statische und dynamische Redundanz oder Zeitredundanz und Diversität. Der letztgenannte Ansatz führt zu den virtuellen Duplexsystemen, die etwa in Lokomotiven eingesetzt werden.

In nur einem einzigen Rechner werden dabei zwei Exemplare eines Prozesses nacheinander ausgeführt (P1 und P2 in Abb. 2) und die Ergebnisse miteinander verglichen. Damit dauerhaft fehlerhafte Hardware nicht zu identisch falschen Ergebnissen führt, werden die Prozesse diversitär ausgelegt, das heißt unterschiedlich programmiert und darüber hinaus durch algorithmische Transformation noch stärker diversifiziert. Dazu wurden verschiedene Verfahren der systematischen Diversität entwickelt. Außerdem sind die Eingaben für P2 durch Informationsredundanz vor Verfälschung zu schützen, während P1 rechnet. Ebenso sind die Ergebnisse von P1 zu schützen, während P2 rechnet. Abbildung 2 zeigt die Stellen, wo Informationsredundanz

hinzugefügt (Dreiecke) und geprüft wird (Rechtecke). Damit ein fehlerhaftes Prozessexemplar die geschützte Information des anderen Exemplars nicht unerkannt verfälschen kann, werden verschiedene Redundanzfunktionen (f und g in Abb. 2) eingesetzt, wofür spezielle diversitäre Codes entwickelt wurden. Sobald die interne oder externe Prüfung eine Abweichung feststellt, wird das System in einen sicheren Zustand überführt. Bei Lokomotiven bedeutet dies eine Notbremsung.

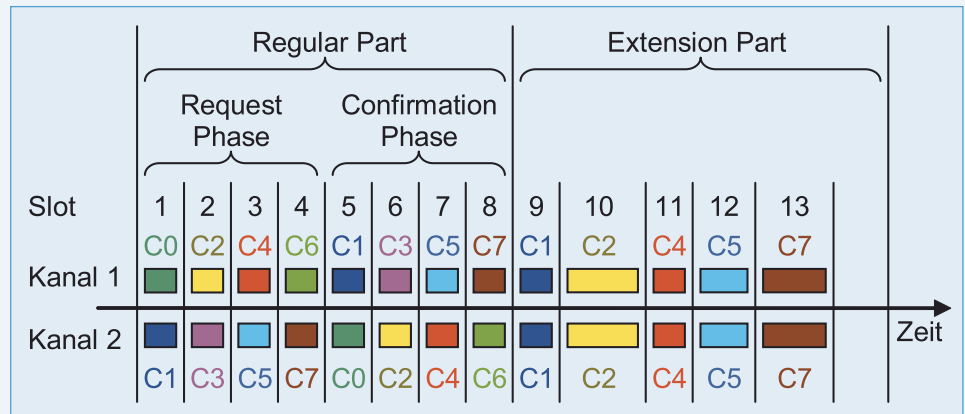


Abbildung 3: TEA-Protokoll

Fehlertolerante Protokolle

Hochgradig fehlertolerante Protokolle sind typischerweise noch redundanter ausgelegt als fehlertolerante Rechner. Deshalb wurde eine Reihe spezieller Protokolle entwickelt, die bei geringem Aufwand eine hohe Zuverlässigkeit erreichen – etwa das Pendelprotokoll, das Network Membership Protocol und das Veto-Protokoll. Speziell für sicherheitsrelevante Aufgaben im Automobil geht der Trend zu dem zeitgesteuerten Kommunikationssystem FlexRay[®]. Es soll bei hoher Bandbreite die einzelnen elektronischen Systemkomponenten von Autos verbinden. Neben der zeitgesteuerten Vergabe der Kommunikationskanäle in Zeit-Slots fester Länge bietet FlexRay[®] noch ein dynamisches Zeit-Segment, in dem sich sendewillige Rechner um die Kanäle bewerben können. Dieses Segment ist jedoch nicht vollständig fehlertolerant ausgelegt. Im neu entwickelten TEA-Protokoll befolgen die Rechner im statischen „Regular Part“ (Abb. 3) eine bestimmte Sendereihenfolge auf zwei Kanälen und tauschen dabei auch ihre Sendewünsche für den dynamischen „Extension Part“ aus. Dafür genügen wenige Bits, die an die Nachrichten angehängt werden. Ein spezieller Fehlertoleranzalgorithmus kann daraus den dynamischen Kanalzugriff im „Extension Part“ ableiten, so dass in TEA die Kommunikation in beiden Teilen eines Kommunikationszyklus durch Fehlertoleranz geschützt ist.

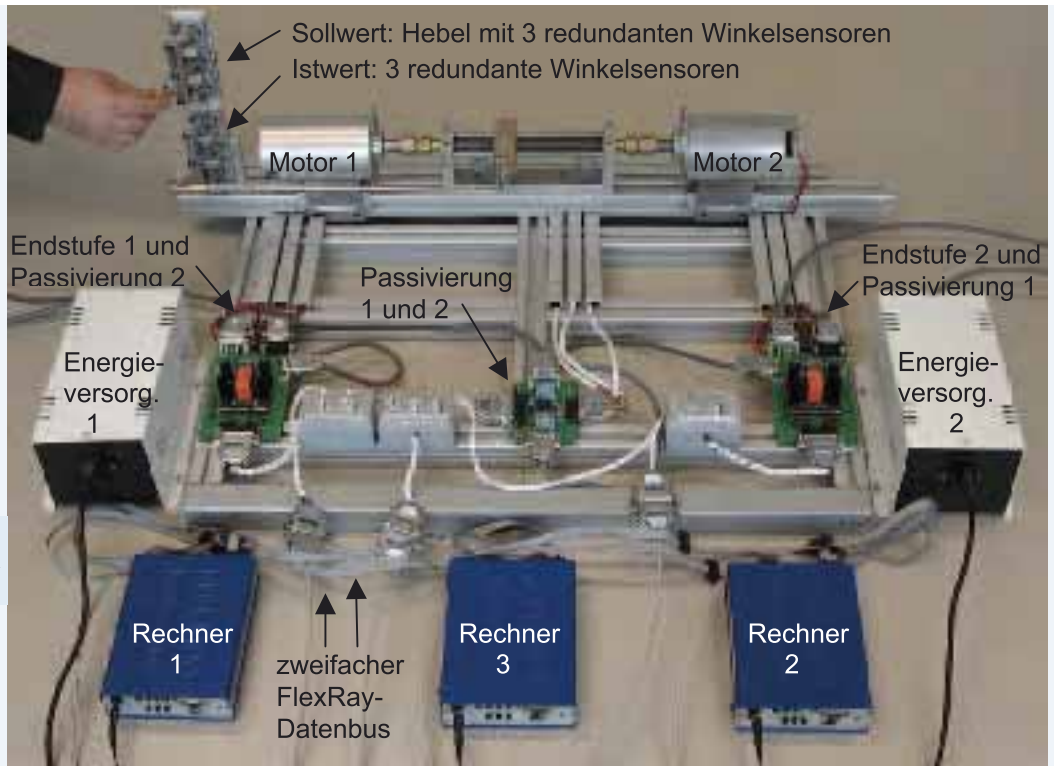


Abbildung 4:
Demonstrator eines fehlertoleranten
Steer-by-Wire-Systems

Mechatronische Systeme

Gemeinsam mit der Automobilindustrie wurden in der Arbeitsgruppe „Verlässlichkeit von Rechen-systemen“ verschiedene fehlertolerante Lösungen für Steuerungs- und Regelungsanwendungen entwickelt. Mit dem Institut für Fahrzeugtechnik des TÜV Nord wurde im Projekt FlexBeam ein Demonstrator für ein fehlertolerantes Steer-by-Wire-System, also eine elektronische Fahrzeuglenkung, entwickelt (Abb. 4). Durch strenge Trennung von Fehlerbereichen entstand ein System, das nachweislich den Ausfall einer beliebigen Komponente toleriert. Bis auf den Stellhebel sind darin auch die Ausfälle der mechanischen Komponenten und der Verkabelung eingeschlossen. Fehlertolerante Lösungen von mechatronischen Systemen wurden auch für den Demonstrator eines fehlertoleranten Steuerungssystems für Hochauftriebshilfen – das sind zum Beispiel Landeklappen – bei Luftfahrzeugen konzipiert.

In dem EU-Projekt EASIS wird gemeinsam mit europäischen Automobilherstellern ein neuer Weg zur effizienten Fehlertoleranz verfolgt. In modernen Fahrzeugen gibt es eine große Zahl von Steuergeräten, die nicht alle voll ausgelastet sind. Daher wird versucht, diese freie Rechenkapazität zur Schaffung von sozusagen „kostenloser“ Redundanz zu nutzen. Dies funktioniert aber nur, wenn eine entsprechende Softwarestruktur geschaffen wird. Vor allem aber benötigt man spezielle fehlertolerante Protokolle, die die primären Rechner und die „entfernte Redundanz“ korrekt und effizient verbinden.

Genetische Algorithmen

Redundante Strukturen können auf vielfältige Weise geschaffen werden. Hohe Fehlertoleranz

bei gleichzeitig geringen Kosten lässt sich oft nur erreichen, wenn man die Redundanzstruktur an spezielle Eigenschaften eines gegebenen Systems anpasst. Diese Entwurfsaufgabe wird in einem neuen Forschungsansatz nicht dem Menschen überlassen, sondern durch einen genetischen Algorithmus automatisiert. Analog zu Vorgängen in der Natur beginnt dieser Ansatz mit einigen primitiven Systemstrukturen und bildet daraus fortgesetzt weitere Generationen von Systemen, beispielsweise durch Mutation und Cross-Over (siehe Abb. 5, links die „Eltern-“, rechts die „Kinder-Systeme“). Eine Fitness-Funktion bewertet jeweils die erreichte Fehlertoleranz und sorgt dafür, dass schlechte Systeme ausgesondert werden und sich allmählich die besseren durchsetzen. Momentan wird dieser Ansatz auf fehlertolerante mechatronische Systeme angewandt.

Sicherheit auf dem Prüfstand

Murphys Gesetz besagt: Alles, was schief gehen kann, geht schief. Tatsächlich ist es noch schlimmer: Alles, was schief gehen kann, geht meistens gut. Folge: Zunächst gute Erfahrungen mit einem System wiegen einen in trügerischer Sicherheit. Aber erst wenn Fehler auftreten, kann man die Wirksamkeit von Schutzmaßnahmen wahrnehmen. Folglich muss man vorgehen wie die Feuerwehr: Sie rückt nicht nur im Brandfall aus, sie macht auch Probealarme und Löschübungen. Aus dem gleichen Grund ist die Fehlertoleranz eng verknüpft mit einer Reihe von Analysemethoden: Hierzu zählen sowohl Experimente unter Injektion von künstlichen Fehlern als auch modellbasierte Untersuchungen wie die Simulation und die Erreichbarkeitsanalyse.

Idealerweise liefern Analysen quantitative Aussagen über die Zuverlässigkeit, beispielsweise:

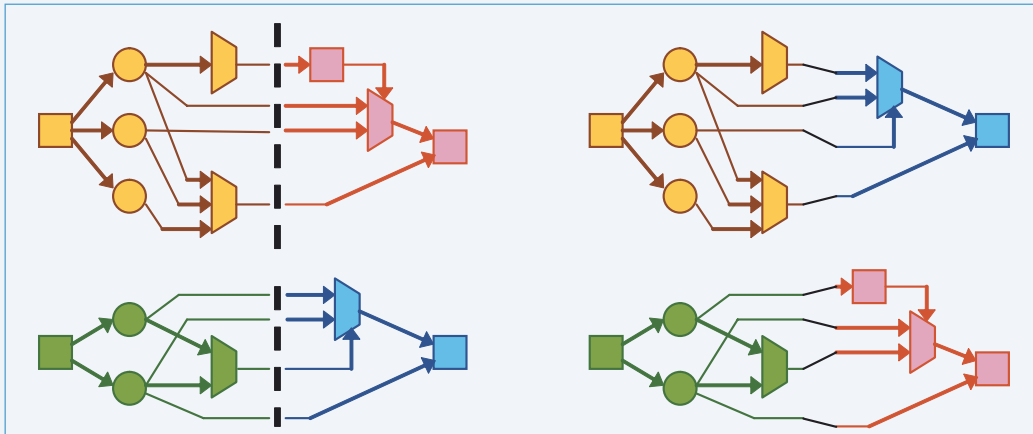


Abbildung 5: Beim Cross-Over werden zwei fehler-tolerante Systeme geteilt (links) und durch Vertauschen der „rechten Hälften“ zu neuen Systemen zusammengesetzt (rechts)

Ein System überlebt einen 30-tägigen Einsatz mit einer Wahrscheinlichkeit von 0,9999999. Derartige Berechnungen verlangen jedoch die Kenntnis von Zuverlässigkeitsdaten der verwendeten Hardware- und Softwarekomponenten. Diese sind oft schwer erhältlich und auch ungenau, da die Komponenten – vereinfacht gesagt – viel schneller unmodern werden als sie ausfallen. Ein Halbleiter-Chip lebt rund 100 Jahre. Deshalb geht man häufig zu einer n-Fehler-Annahme über und verlangt von einem System, dass es bis zu n auftretende Fehler tolerieren muss. Diese Eigenschaft ist anhand eines Verhaltensmodells von fehlerfreien und fehlerhaften Komponenten zu beweisen oder durch Fehlerinjektionsexperimente statistisch zu zeigen.

Computergestützte Analyse

Wenn die Erfüllung einer n-Fehler-Annahme zu prüfen ist, geht man oft von Zustands-Transitions-Modellen aus – beispielsweise von einer Menge endlicher Automaten, die je eine Komponente repräsentieren. Jeder Automat beschreibt auch das zeitliche Verhalten und seine Wechselwirkungen mit den anderen Automaten. Daraus können durch Erreichbarkeitsanalyse alle Möglichkeiten abgeleitet werden, wie sich das Gesamtsystem verhalten kann. Aufgrund der vielen Fehlermöglichkeiten und zeitlichen Variabilitäten entstehen leicht extrem viele Verhaltensweisen, so dass nur ein Programm (Model Checker) die Analyse durchführen und dabei prüfen kann, ob alle Fehler toleriert werden. Mit dieser Methode wurde in einem mehrjährigen Projekt mit dem FlexRay-Konsortium das FlexRay®-Kommunikationssystem für Automobile geprüft.

In einem laufenden Forschungsvorhaben werden Ansätze entwickelt, um die Analysedauer beträchtlich zu verkürzen. Dazu fließt Fehlertoleranz-Wissen in den Überprüfungs-Algorithmus ein,

beispielsweise die Unterscheidung von Fehlerbereichen („single fault region“ in Abb. 6) sowie eine Klassifikation der Transitionen („LIT“ und „GT“). Bestimmte lokale Vorgänge können dann lokal betrachtet werden, ohne Wechselwirkungen mit anderen Bereichen berücksichtigen zu müssen. Durch zusätzliche Heuristiken können Verletzungen der Fehlertoleranz noch schneller aufgespürt werden. Ebenso kann die Analyse vereinfacht werden, wenn die periodische Arbeitsweise in stets gleich langen Zyklen bei der Modellbildung berücksichtigt wird.

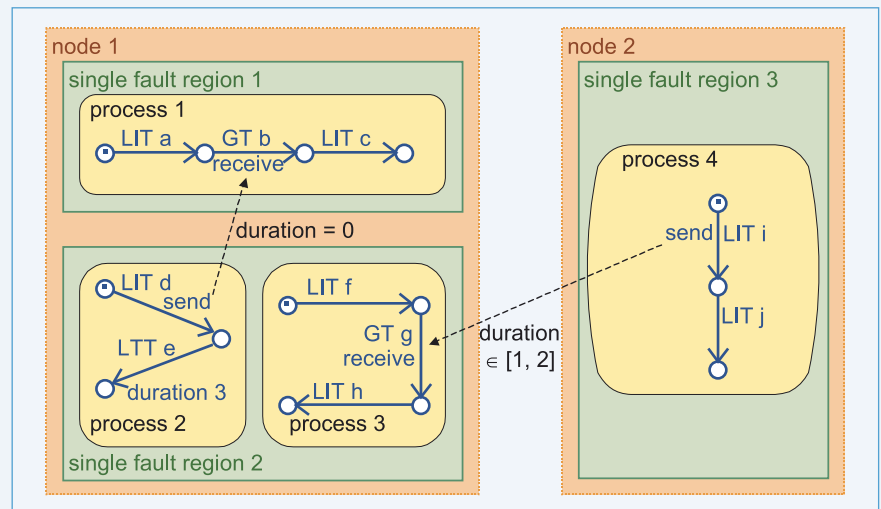


Abbildung 6: Effiziente Erreichbarkeitsanalyse eines fehler-toleranten Systems

Fehlerinjektion

Neben dem modellbasierten wird auch das experimentelle Vorgehen erforscht. Durch eine Reihe von Fehlerinjektoren, die ausschließlich durch Software implementiert sind, wurde in vielen Systemen die Möglichkeit geschaffen, künstlich Fehler in ein System einzubringen und so die Reaktion des Fehlertoleranzverfahrens zu testen. Generell besteht die Schwierigkeit, aus der Vielzahl der Fehlermöglichkeiten die „interessanten“ Fehlerfälle herauszusuchen, für die man je ein



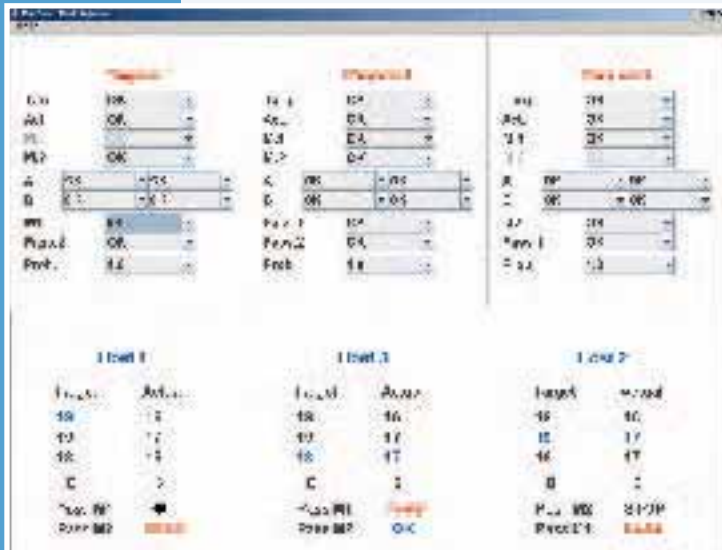


Abbildung 7:
Injektion des Fehlers
„Motor 1 wird fälschlicher-
weise mit einem Signal zur
Linksrotation angesteuert“.

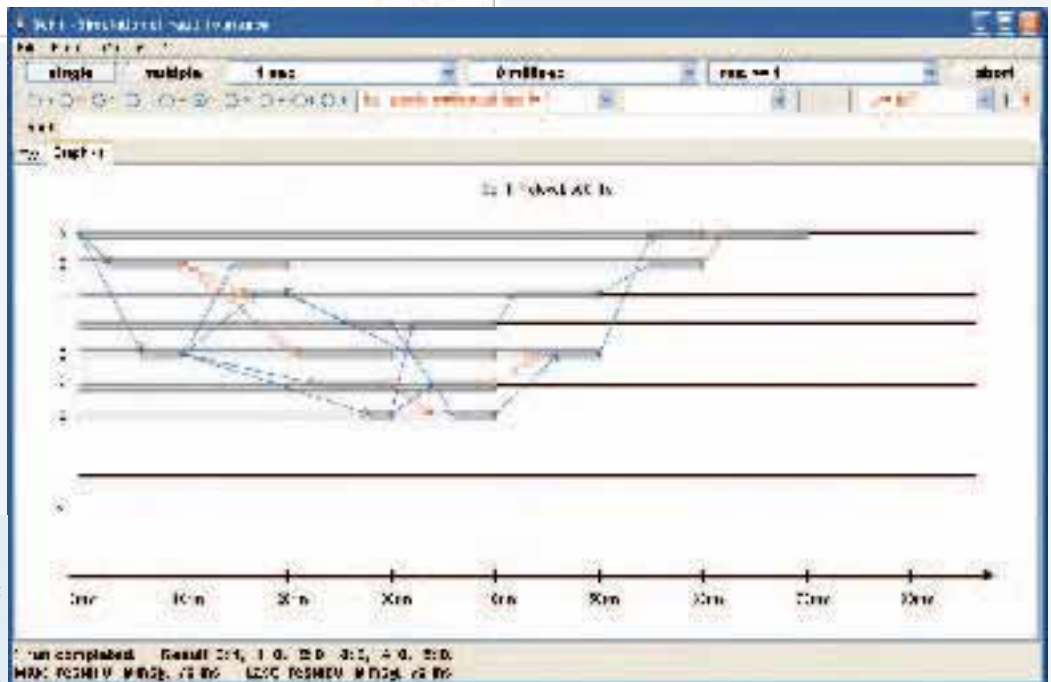


Abbildung 8:
Das Tool SoFT (Simulation
of Fault-Tolerance) visualisiert
die Kommunikation zwischen
Rechnern, wobei in Rechner B
und F Fehler injiziert worden sind

Injektionsexperiment durchführt. Anders als beim Test von mechanischen Komponenten kommt es bei der Fehlerinjektion nicht darauf an, möglichst schwerwiegende Fehler zu injizieren. Diese sind meist so offensichtlich, dass ein Rechensystem sie leicht zu erkennen und zu tolerieren vermag. Viel entscheidender sind dagegen kleine, subtile Zeit- oder Datenveränderungen, die unbemerkt bleiben können. Bei manchen der entwickelten Fehlerinjektoren gibt der Mensch die Fehler vor, bei anderen werden diese anhand eines Modells berechnet.

Im FlexBeam-Projekt mit dem Institut für Fahrzeugtechnik des TÜV Nord wurde ein Fehlerinjektor entwickelt, der sehr hardwarenah arbeitet, um die Verfälschung der Injektionsergebnisse durch den Injektor selbst möglichst gering zu halten. Abbildung 7 zeigt die Bedienerschnittstelle, die es ermöglicht, für alle Komponenten eines mechatronischen Systems (siehe Abb. 4) Fehlfunktionen festzulegen.

Speziell für die universitäre Lehre wurde ein Simulationssystem entwickelt (Tool SoFT, siehe Abb. 8), mit dem Studierende selbst Fehlertoleranzverfahren programmieren und unter Fehlerinjektion erproben können. Durch eine graphische Ausgabe werden die von Fehlern hervorgerufenen Verhaltensweisen des Systems sichtbar gemacht.

Die Entwicklung von Sicherheitskonzepten für Rechensysteme bedingt ein ausgewogenes Verhältnis von Zuverlässigkeit und Redundanzaufwand, um ein hohes Maß an Sicherheit mit einem vertretbaren wirtschaftlichen Auf-

wand zu erreichen. Hierbei haben sich Kombinationsmodelle bewährt. Die Zuverlässigkeit sicherheitstechnischer Systeme bedarf der sorgfältigen experimentellen, modellbezogenen und computergestützten Überprüfung, um dauerhaft optimale Verlässlichkeit gewährleisten zu können.

Kontakt

Prof. Dr. Klaus Echtle

Verlässlichkeit von Rechensystemen

Tel.: 02 01 / 1 83 - 23 52

Fax.: 02 01 / 1 83 - 46 98

echtle@dc.uni-due.de

http://dc.uni-due.de

| Software-Ingenieure · IT-Architekten · IT-Projektleiter · IT-Consultants* |
+++ 300 neue, spannende Jobs in 2006 für Einsteiger und Profis +++



I II

Spannung gehört zu unseren Aufgaben. Entspannung gehört zu unserer Kultur.

sd&m ist eines der renommiertesten Software- und Beratungshäuser mit über 1.000 hoch qualifizierten IT-Fachleuten. Neben einem attraktiven Kundenportfolio, vielen erfolgreich durchgeführten Projekten und einem gesunden Wachstum sind wir vor allem auf eines stolz: auf unsere besondere, eigenständige Unternehmenskultur. Ein Arbeitsklima, das auf Kooperation, Teamgeist und Motivation beruht, trägt maßgeblich zum Erfolg unserer Projekte bei. Aufgrund der anhaltend guten Auftragslage suchen wir neue Kolleginnen und Kollegen für alle Niederlassungen.

Als **Software-Ingenieur** sind Sie verantwortlich für herausfordernde Software-Engineering-Aufgaben. Im Team spezifizieren, entwerfen, programmieren, testen und integrieren Sie maßgeschneiderte Lösungen direkt für unsere Kunden und arbeiten an Studien. Mit einiger Erfahrung übernehmen Sie zu-

sätzlich mehr eigene Verantwortung und leiten kleinere Projekte.

Als **IT-Architekt** sind Sie verantwortlich für die Anwendungsarchitektur oder die technische Architektur der IT-Lösung für einen Kunden in einem konkreten Projekt. Innovative Konzepte wie serviceorientierte Architektur, komponentenbasierte Architektur oder MDA wollen wir praktisch nutzbar machen. Als Architekt steuern Sie ein Team von bis zu 30 Mitarbeitern in der Rolle des Chefdesigners. Sie beraten und coachen den Projektmanager und führende Vertreter des Kunden. Sie bringen Ihre Architekturkompetenz über Communities unternehmensweit ein.

Als **IT-Projektleiter** verantworten Sie ein Projekt als Ganzes. Sie sind der erste Ansprechpartner für unseren Kunden und für die Leitung des Geschäfts-

bereichs. Sie führen das Projektteam, Sie verantworten Budget, Qualität und Termine. Sie halten Balance zwischen strategischen und fachlichen Anforderungen des Auftraggebers, den Wünschen und Bedürfnissen des Anwenders, den Budget- und Terminvorgaben, den Qualitätsansprüchen des Teams und Ihren eigenen Wertvorstellungen. Durch exzellente Projektdurchführung erzielen Sie hohe Kundenzufriedenheit und können so neue Aufträge gewinnen. Ihr Engagement hilft sd&m als Unternehmen, gegen den Wettbewerb weiter zu wachsen.

Als **IT-Consultant** innerhalb der IT-Beratung von sd&m übernehmen Sie anwendungsübergreifende Aufgaben zur IT-Architektur, zur Betriebsführung, zu Geschäftsprozessen und zum Projektmanagement. Sie übernehmen Führungsaufgaben und arbeiten an der inhaltlichen und geschäftlichen Weiterentwicklung unserer IT-Beratung.

Wir freuen uns auf Ihre Bewerbung mit Zeugnissen, in der Sie uns zeigen, was Sie können. Sie finden uns in **München, Stuttgart, Frankfurt, Köln/Bonn, Düsseldorf, Hamburg, Berlin und Zürich**. Bitte bewerben Sie sich in der Niederlassung Ihrer Wahl. Mehr zu Projekten, Team, Kultur und alle Kontaktinformationen finden Sie unter: www.sdm.de

* m/w



sd&m
A Company of Capgemini