



ILLUSTRATION: MICHAEL STACH



Der Geist in der Maschine

DIE WELT IST VOLLER
GEFAHREN. DOCH
MASCHINEN, DIE IHR
EIGENES INNENLEBEN
KENNEN, KÖNNEN KRISEN
SELBST BEWÄLTIGEN
– UND ZWAR BESSER
ALS MENSCHEN.

VON WADE ROUSH

Der „Mars Polar Lander“ hatte keine Chance. Kurz nach Mitternacht am 3. Dezember 1999, nach einer elfmonatigen Reise, schwenkte die Marssonde der NASA ihre Antenne von der Erde weg, um den Eintritt in die Mars-Atmosphäre vorzubereiten. Das war das Letzte, was das Kontrollzentrum der US-Weltraumagentur von ihr mitbekam. Der Rest ist Rätsel. Wahrscheinlich verließ die Sonde ihre Umlaufbahn, öffnete ihren Fallschirm und zündete die Bremsraketen, um den Fall weiter zu verlangsamen – alles genau nach Plan. Doch dann meldeten die Sensoren an den drei Beinen der Fahre der Steuerungssoftware viel zu früh Bodenberührung. Auf diese Situation war das Programm nicht vorbereitet: 40 Meter über dem Marsboden schalteten sich die Bremsraketen ab – andere Messungen, nach denen die Sonde noch nicht aufgesetzt hatte, wurden ignoriert. Die Schwerkraft tat ihr Werk, und die empfindliche Sonde stürzte mit hoher Geschwindigkeit auf den felsigen Marsboden.

Im gleichen Jahr, Millionen von Kilometern entfernt, war ein anderes Raumschiff der NASA geschickter bei der Krisenbewältigung: „Deep Space One“ hatte gerade eine Drehung begonnen, um den Vorbeiflug an einem nahen Asteroiden vorzubereiten. Weil die Sonde ihre Energie dabei dringend für andere Komponenten brauchte, hatte ihre Kamera abgeschaltet werden müssen. Doch der Schalter klemmte, und die Sonde drohte in einen quasi bewusstlosen, „fehlersicheren“ Zustand zu fallen. Den wieder aufzuheben, hätte die Bodenkontrolle Wochen gekostet. Der Asteroid wäre dann längst außer Sicht gewesen.

Doch Deep Space One hatte etwas, was der Polar Lander nicht hatte: einen integrierten Roboter, der selbstständig denken und deshalb auch mit der unerwarteten Situation umgehen konnte. Mit seinen eigenen technischen Kenntnissen unternahm der Roboter zunächst einen Reparaturversuch, indem er mehrfach probierte, den verhakten Schalter aus- und wieder einzuknippen. Als dies fehlschlug, entwickelte er eigenständig einen Alternativ-Plan, um die vorgesehene Kurskorrektur zu Ende zu bringen.

Der funktionierte, und die Sonde konnte ihre Mission fortsetzen.

MEHR HIRN ALS MUSKELN

Der Roboter, der Deep Space One aus der Patsche half, gehört zu den ersten Vertretern einer völlig neuen Gattung. Anders als die Metallfiguren, die steif durch Science-Fiction-Filme marschieren, oder ihre abgespeckten Verwandten, die am Fließband Teile zusammenschweißen, haben die neuen Roboter mehr Hirn als Muskeln. Sie verfügen über ein in Software gegossenes detailliertes Bild von ihrem Aufbau und ihren Funktionen. Dadurch können sie auch auf Ereignisse reagieren, die ihre Programmierer nicht vorhergesehen haben. Weil viele dieser Maschinen unbeweglich sind, werden sie „Immobil Roboter“ oder kurz „Immobots“ genannt. Sie könnten bedeutende Veränderungen auslösen – im All wie auf der Erde.

Immobots tauchen schon heute in Anlagen auf, bei denen es auf selbstständiges Handeln ankommt. Benötigt werden sie zum Beispiel, wenn eine direkte Steuerung durch den Menschen nicht möglich ist – wie eben in einer

Raumsonde, die so weit von der Erde entfernt ist, dass Funksignale Minuten oder gar Stunden brauchen, um sie zu erreichen. Oder weil Menschen die Fähigkeit oder die Lust abgeht, alle Details zu überwachen. „Es gibt auf der Welt eine große Zahl von Systemen mit Sensoren und Reglern, die völlig anders aussehen als herkömmliche, bewegliche Roboter“, sagt Brian Williams, ein ehemaliger NASA-Forscher, der an der Entwicklung der Software für Deep Space One beteiligt war. Heute ist er Professor am Labor für Raumfahrtssysteme und künstliche Intelligenz des Massachusetts Institute of Technology (MIT).

„Sobald ein System mit der Immobot-Software programmiert ist, hat es eine dem gesunden Menschenverstand entsprechende Vorstellung von seinen eigenen Komponenten. Aus diesem Modell kann es herleiten, was nicht in Ordnung ist, und erkennen, was es dagegen tun kann“, sagt Williams. Normale Computer sind gut im schrittweisen Abarbeiten von stur zu befolgenden Instruktionen, die ihnen von Programmierern vorgegeben werden. Geht es jedoch darum, eine unvorhergesehene Situation zu meistern,

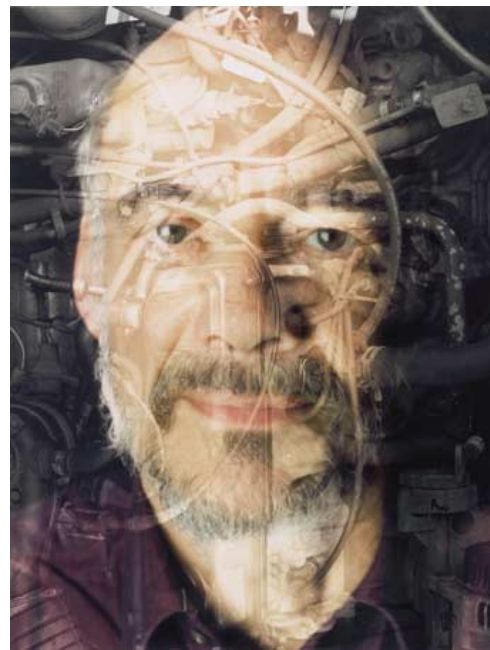
wie sie etwa beim Versagen eines Bauteils vorliegt, dann sind diese Computer hilflos.

Von der Vorgehensweise nach Immobot-Art profitieren immer häufiger ganz alltägliche Systeme wie etwa Autos, Bürokopierer oder Datenbanken. Wenn die Entwicklung so weitergeht, dürften bald auch grundlegende Infrastrukturen wie Heizung und Belüftung, Telefonsysteme und Computernetze, die Luftverkehrsüberwachung und Stromnetze durch Immobot-Gehirne gesteuert werden. Auf lange Sicht könnten sich ganze Städte und Regionen zu verteilten Immobots entwickeln.

BESSER ALS SCHACH

Die Leistung von Deep Space One in einer schwierigen Situation war für die neue Rasse der Immobots ein früher Triumph. Aber in Wirklichkeit stellte der blockierte Schalter keine echte Krise dar: Die Bodenkontrolle hatte das Steuerprogramm der Sonde absichtlich in die Irre geführt. Die Techniker wollten prüfen, wie die von Williams und seinen NASA-Kollegen entwickelte Software „Remote Agent“ reagieren würde. Alles war Teil eines umfangreichen Feldversuchs mit einer neuen Programmierphilosophie, die als modellbasiertes logisches Denken bezeichnet wird und das Grundprinzip jeder Immobot-Entwicklung darstellt.

Die Basis für – mehr oder weniger – autonome Maschinen sind in der heutigen Praxis meist so genannte „heuristische“ Programme. Um ihre Ziele zu erreichen und sich gegen alle Eventualitäten zu wappnen, arbeiten sie mit Regellisten nach dem Muster: „Wenn A wahr ist, dann tue B. Wenn C wahr ist, dann tue D.“ Doch Experten für künstliche Intelligenz müssen einräumen: Solche von Hand programmierte Software ist entweder zuverlässig oder bezahlbar – niemals beides zugleich. Sie muss – siehe Polar Lander – mit außerordentlich komplexen und oft überraschenden Situationen umgehen können. Menschliche Programmierer mit ihren knappen Terminen und begrenzten Budgets können kein Programm schreiben, das jede denkbare Möglichkeit vorhersieht. Versuchen sie es trotzdem, entsteht ein überfrachtetes



Forscher Struss: Entwickelt intelligente Diagnosesysteme für Autos und Wasserwerke.

und schludrig geschriebenes Werk mit versteckten und möglicherweise fatalen Fehlern.

Modellbasierte Programme wie der Remote Agent der NASA sind völlig anders aufgebaut. Sie sind Abbilder der Maschinen, die sie steuern sollen, gezeichnet in der logischen Sprache eines Computers. Ausgerüstet mit diesem Bauplan können sie eigenständig den schnellsten, sichersten und kostengünstigsten Weg wählen, um den Befehl des Operators umzusetzen oder eine gefährliche Störung zu beheben. „Die Idee ist ganz einfach“, sagt Williams,

„Gib dem Programm einen physischen Plan des Systems und lass die Software ermitteln, was zu tun ist.“

Der Schlüssel zur Zuverlässigkeit des Remote Agent im Praxistest lag in der Kombination von mehreren sehr einfachen Modellen, erläutert Robert Rasmussen, Chefkonstrukteur des Projektes „Mission Data System“ am Jet Propulsion Laboratory der NASA. Jedes davon befasst sich nur mit einer der mechanischen oder elektrischen Komponenten der Sonde. Für ein Ventil etwa gibt es zwei Betriebszustände – „offen“ oder „geschlossen“ – und zwei

Modellbasierte Software: Autos mit Gespür

Peter Struss hat sein Urlaubsabenteuer glimpflich überstanden: Am letzten Februartag war der Münchner Wissenschaftler und Softwareunternehmer in Kambodscha von einer überfüllten Fähre in den Mekong-Fluss gefallen. „Am Ufer stellte ich fest, dass mein linker Fuß schräg wegsteht“, erinnert er sich. „Ich klopfte ihn wieder gerade, aber nach ein paar Schritten war er wieder schief. Da wusste ich, dass es etwas Ernstes war.“ In Kambodscha bekam er keine adäquate medizinische Versorgung, also schlug er sich, hüpfend auf dem rechten Bein, bis nach Deutschland durch. Nun liegt Struss mit eingegipftem Sprunggelenk im Münchner Klinikum Rechts der Isar, schon wieder umgeben von Laptops und Papierstapeln.

Von der ersten Schadensmeldung über die Analyse des Defekts zu den richtigen Gegenmaßnahmen: Mit den von Struss und seinem Partner Oskar Dressler entwickelten Programmen sollen Maschinen in unerwarteten Krisensituationen ebenso souverän reagieren. „Wie nützlich dies wäre, kann jeder bei einer Autopanne erleben: Woran liegt es, wenn schwarzer Qualm aus dem Auspuff dringt? Schwächelt die Einspritzpumpe? Ist eine Zündkerze defekt? Der Turbolader leckt? Oder spinnt die Elektronik? Der Kennerblick unter die Motorhaube führt da kaum weiter. Moderne Autos sind längst fahrende Computernetzwerke, mit

etwa hundert Prozessoren. Selbst ausgebildete Mechaniker und Ingenieure haben Probleme, das Verhalten eines derart komplizierten Systems in allen Einzelheiten zu durchschauen. Struss' Lösung heißt „modellbasierte Diagnose“: Der Hersteller legt eine Bibliothek formalisierter Funktionsbeschreibungen („Modelle“) aller Bauteile eines Autos an. Daraus und aus elektronischen Konstruktionsplänen setzt das Programm Raz! von Struss' Firma Occ'n! dann automatisch Modelle der Subsysteme des Autos zusammen – etwa des Antiblockiersystems, des Turboladers oder der Klimaanlage.

Gefüttert mit so einem Modell kann ein Rechner nachspielen, wie sich das System verhält, welche Symptome ein bestimmter Fehler bewirkt und wie dieser sich durch das System fortplant. An Bord des Autos und mit geeigneten Sensoren verbunden, kann der Rechner selbstständig Defekte diagnostizieren. Das Auto bekommt damit eine Art Gespür für Probleme und kann entsprechend reagieren: bei einem Druckabfall in der Bremsflüssigkeit beispielsweise den Motor drosseln. Dem Mechaniker sagt es dann, welches Teil er auswechseln muss – oder es schlägt ihm bei mangelnder Datenlage Tests vor, um den Fehler einzugrenzen. „Eines allerdings hat die Natur der Technik noch voraus: Das Sprunggelenk von Peter Struss heilt von allein. Hier kein Auto kann eine defekte Zündkerze auswechseln. — Tobias Hüter



Deep Space One: Der eingebaute Immo-
bot half der Sonde aus der Patsche.

Fehlerzustände, nämlich „offen blockiert“ oder „geschlossen blockiert“. Mathematische Regeln beschreiben die möglichen Übergänge zwischen diesen Zuständen und die zugehörigen Wahrscheinlichkeiten. So ist es höchst unwahrscheinlich, dass ein Ventil von „offen blockiert“ in den Zustand „geschlossen blockiert“ wechselt. Im Fehlerfall weiß das Programm dann, dass es sich bei der Untersuchung dieser Möglichkeit kürzer fassen kann.

Im nächsten Schritt wurden Hunderte von diesen Untermodellen getreu den Konstruktionsplänen von Deep Space One zu einem Gesamtbild der Sonde zusammengefasst. Im realen Betrieb kann das Kontrollzentrum ihr einfach ein bestimmtes Ziel vorgeben. Der ImmoBot macht sich dann ein Bild vom momentanen Zustand der Sonde, das sich aus den Sensoren ergibt, die jedes Ventil, jedes Relais, jeden Kreiselkompass, jeden Treibstofftank und jede Kamera überwachen. Mit dem aktuellen Lagebild und mit Hilfe seines Wissens über das Innenleben der Sonde kann er einen Plan entwickeln, wie er Schritt für Schritt dem Ziel näher kommt – und Probleme auf dem Weg dahin umgehen oder lösen.

„Wollte man als Techniker alle Schwierigkeiten vorhersehen und dafür

sorgen, dass die Sonde dabei richtig reagiert, wäre dies wie in einem extrem zeitaufwendigen Schachspiel“, sagt Chefentwickler Rasmussen. Anders bei der modellbasierten Programmierung: „Damit können wir das gewünschte Systemverhalten herbeiführen, ohne dass wir dafür Jahre und Millionen von Dollars investieren müssen.“

DAS AUTONOME BÜRO

Eine Raumsonde muss eindeutig mehr können als beispielsweise ein Kopierer. Aber auch stark belastete Büromaschinen müssen, was ihre Zuverlässigkeit und Effizienz anbelangt, hohen Anforderungen genügen – zum Beispiel, wenn der Chef 30 gebündelte Ausgaben des Jahresberichts für eine Vorstandssitzung verlangt, die in drei Minuten beginnt. Deshalb haben Techniker bei Rank Xerox und im vor kurzem ausgegliederten Forschungszentrum Palo Alto Research Center (PARC) damit begonnen, ihre Spitzen-Kopiermaschinen mit der Intelligenz von ImmoBots auszustatten.

Die größte Herausforderung für einen Kopierer liegt wie bei einem Großflughafen mit wenigen Rollbahnen in der Planung: Das nächste leere Blatt

DIE IDEE IST GANZ EINFACH: GIB DEM PROGRAMM EINEN PHYSISCHEN PLAN DES SYSTEMS UND LASS DIE SOFTWARE ERMITTELN, WAS ZU TUN IST.

Papier muss dem Drucksystem zugeführt werden, sobald das vorherige aus dem Weg ist. „Als die früheren Generationen von Xerox-Kopierern und Druckern immer komplexer wurden, fiel auf, dass dieser Steuerungsvorgang mehr und mehr Zeit benötigt, und zwar besonders bei Optionen wie doppelseitigem Kopieren, Sortieren oder Heften“, sagt Daniel Bobrow, Forscher für künstliche Intelligenz am PARC-Labor für Systeme und Verfahren.

Obendrein fehlte der heuristischen Steuerung das rechte Verständnis für die Interaktion zwischen den Stationen. Sie behandelte jeden Prozess als separate Aufgabe, die erledigt werden musste, bevor das nächste Blatt auf den Weg gebracht werden konnte. Und jedes Mal, wenn Xerox einen Kopierer mit neuen Funktionen ausstattete

wollte, musste die Steuerungssoftware von Grund auf umgeschrieben werden.

Heute verkauft Xerox mit dem DocuColor 2045 und dem DocuColor 2060 zwei Druckmaschinen, die einen großen Schritt in Richtung Intelligenz gemacht haben. Die beiden lastwagen-großen Geräte in der Preisklasse um 100 000 Euro arbeiten mit modellbasierten Steuerprogrammen, die den Papierfluss ständig optimieren.

Angenommen, von einer Broschüre sollen 70 Kopien erstellt werden. Nach vor dem Drücken der Start-Taste weiß eine Maschine nach dem PARC-Modell, dass dabei am meisten Zeit für das Heften benötigt wird. Diese Information gibt sie an andere Komponenten weiter und erlaubt ihnen dadurch, parallel so zu arbeiten, dass kein großer Rückstau entsteht. Wie bei der NASA-Sonde setzt sich das Software-Bild eines ganzen DocuColor-Geräts aus vielen kleineren Modellen von Einzelkomponenten zusammen. Rüstet man eine neue Komponente wie etwa einen Scanner oder einen Sortierer nach, überträgt sie ihr internes Modell an die zentrale Steuerung und fügt sich so nahtlos ein – quasi ein automatisches Software-Upgrade ganz ohne Programmierarbeit.

„Für die Steigerung der Produktivität ist das modellbasierte Programmieren eine fantastische Sache“, schwärmt Bobrow. Nutzer könnten Komponenten nach Bedarf hinzufügen oder entfernen und ihr Gerät stundenlang unbeaufsichtigt arbeiten lassen. Noch besser sei, dass Xerox seine aktuellen Geräte jetzt nutzen könne, um mit Hilfe der Modelle neue Gerätekonfigurationen zu simulieren und zu bewerten, „bevor nur ein Cent in die physischen Prototypen der nächsten Maschinen-Generation gesteckt wird.“

LERNFÄHIGE DATENBANKEN

Es ist ein fundamentaler Unterschied, ob man einer Maschine sagt, wie sie ihren Job machen soll, oder ob man einem System nur das



Schlaue Autos: Wenn die Klimaanlage nicht funktioniert, kann die eingebaute Kontroll-Software den Fehler finden und vielleicht sogar beheben.

gewünschte Endresultat vorgibt“, sagt Robert Morris, Leiter des IBM-Forschungszentrums Almaden im kalifornischen San Jose. IBM arbeitet daran, Produkte wie Web Server oder Speichernetzwerke autonom zu machen – mit modellbasierter Software wie bei Xerox, aber auch mit klassischer heuristischer Programmierung. In einem der Projekte wird beispielsweise das IBM-Datenbankprogramm DB/2 so ausgebaut, dass es aus früher gestellten Abfragen lernt und neue Formulierungsvorschläge, die schneller und präziser beantwortet werden können (siehe Kasten S. 96).

IN VOLLER FAHRT

Bei Kopierern ist ein geschickter Umgang mit Problemen nützlich – bei Autos ist er lebenswichtig. Und an der modellbasierten Überwachung von Computer-Funktionen im Auto wird mit Hochdruck gearbeitet. Moderne Autos sind mit Dutzenden von Prozessoren – so genannten elektronischen Steuer-einheiten – ausgestattet, die bestimmte Funktionen überwachen und steuern. Mehrere Forschergruppen in Europa sind bereits dabei, integrierte ImmoBots zu konstruieren, die mit den Steuereinheiten zusammenarbeiten und auf Störungen reagieren, sobald – oder sogar bevor – sie auftreten.

Heuristisch angelegte Programme sind bei Problemen oft überfordert, weil sie das Grundlagewissen eines menschlichen Experten haben,

sagt Peter Struss, Computerfachmann an der Technischen Universität München. Mit „Grundlagewissen“ meint Struss die Kenntnisse über Struktur, Verhalten und Zusammenwirken von Ventilen, Schläuchen, Sensoren und anderen Teilen des Fahrzeugs. Bei Occ'm, einer von ihm mitgetragenen Softwarefirma, hat Struss in Kooperation mit großen Auto-Herstellern Softwaremodelle von Autokomponenten entwickelt und getestet. Seine ImmoBots fanden Fehler, die manchmal sogar hochqualifizierten Mechanikern entgehen (siehe Kasten S. 92).

Falls in Ihrem Auto die Klimaanlage ausfällt, kann das an einem defekten Sensor im Benzintank liegen. Warum? „In einigen Fahrzeugen muss die Steuerung der Klimaanlage zunächst die Motorsteuerung fragen, ob sie sich als Verbraucher einschalten darf. Die Motorsteuerung fragt dann ab, ob genug Benzin da ist. Und wenn nicht, wird die Anfrage der Klimaanlage einfach abgelehnt“, erklärt Struss.

Ein modellbasiertes Diagnosesystem kennt diese Details. Wenn der Benzintank voll ist und die Klimaanlage trotzdem nicht funktioniert, dann entwickelt es eine neue Vermutung über die Ursache. Struss: „Der Kfz-Mechaniker in der Werkstatt kann die Diagnosen aus einer Steuerung auslesen und hat dann sechs oder sieben Fehlercodes. Daraus geht aber nicht unbedingt hervor, was die Ursache ist. Die gesamte Interaktion im System bleibt für ihn unsichtbar.“ Für den ImmoBot nicht.

SAUBERES WASSER

Raumsonden, Kopierer und Autos – drei Einsatzbereiche für ImmoBots, bei denen es um vergleichsweise kleine Objekte geht. Wie Struss aber in Brasilien demonstriert, können ImmoBots auch Anlagen von ganz anderer Dimension steuern: Für die Trinkwasserversorgung der Industriestadt Porto Allegre mit ihren 1,3 Millionen Einwohnern entwickelt er ein System, das Wassereigenschaften wie Trübung, Farbe und Säuregrad überwachen soll. Es hat die Aufgabe, bei Problemen früh Alarm auszulösen und automatisch Gegenmaßnahmen vorzuschlagen.

Kein einfaches Unterfangen. Denn ein Wasserspeicher ist ein Ökosystem und keine Maschine, erklärt Struss. „Das bedeutet, dass sich unser Modell nicht allein für die Suche nach fehlerhaften Komponenten eignen muss, es muss auch unvorhergesehene Einflüsse und Interaktionen berücksichtigen.“

Ändern sich beispielsweise die Sensormeldungen, kann eine ImmoBot-gesteuerte Aufbereitungsanlage nur dann auf die wahrscheinlichste Ursache zurückschließen, wenn sie genügend verknüpfte Modelle der physikalischen und biologischen Vorgänge kennt, von denen die Wasserqualität beeinflusst wird. Angenommen, der Eisengehalt des Wassers wäre gefährlich hoch; das System sollte dann als Hypothese eine Algenblüte für das Problem verantwortlich machen, denn absterbende Algen verändern den Säuregrad des Wassers, sodass abgelagertes Eisen wieder gelöst wird. Im Idealfall würde der ImmoBot

den Bedienern kurz- und langfristige Maßnahmen vorschlagen: eine sofortige Wasserbehandlung zur Reduzierung des Eisengehaltes und eine Untersuchung der wahrscheinlichsten Ursache für die Algenblüte, nämlich der Zufuhr von zu viel Düngemitteln aus der Landwirtschaft.

Aber vollständige und exakte Modelle von natürlichen Prozessen zu entwerfen und all diese Modelle störungsfrei zusammenwirken zu lassen, übersteigt nach Struss' Ansicht derzeit noch die Möglichkeiten der Softwareentwickler. „Das ist schwierig, weil man nicht alle beteiligten Faktoren erfassen kann. Manchmal weiß man einfach nicht, woran etwas liegt.“

AUTONOMIE OHNE RISIKO

Während Struss und andere Forscher weiter an den großen Herausforderungen arbeiten, finden die ImmoBot-Programme schnell ihren Weg in kleinere eigenständige Systeme. Aber bevor der Software umfassende Infrastruktur-Technologien anvertraut werden können, müssen die Entwickler noch eine ganz andere Hürde überwinden: die Skepsis

der Ingenieure alter Schule, die ihre Maschinen traditionsgemäß an der kurzen Leine heuristischer Steuerprogramme führen.

Bei der NASA etwa hätte Misstrauen das Projekt des Remote Agent fast gekippt. „Projektleiter sind naturgemäß extrem risikoscheue Menschen“, sagt Ken Ford, Direktor des „Institute for Human and Machine Cognition“ an der Universität von West Florida. Der Computertechniker, der während der Entwicklung des Remote Agent ebenfalls in leitender Position bei der NASA tätig war, hat die Scheu vor dem neuen Vorhaben selbst erlebt: „Es war wirklich schwierig, die Leute dazu zu bringen, es überhaupt zu probieren. Sogar gegen das reine Experiment gab es Widerstand.“

Aber der Erfolg des Remote Agent und andere Demonstrationen der Fähigkeiten modellbasierter Software könnten allmählich auch die Haltung konservativer Techniker ändern. Robert Rasmussen vom Jet Propulsion Laboratory etwa arbeitet mit dem Team von Brian Williams an der Entwicklung eines modellbasierten Programms, das möglicherweise als Hauptsteuerungssoftware für das „Mars

Science Laboratory“ dienen wird, das 2009 abheben soll. Im schottischen Edinburgh verkauft eine kleine Firma in Kooperation mit dem US-Konzern General Electric bereits modellbasierte Software für die Überwachung des Gasturbinenverhaltens bei der Stromerzeugung. Und die Europäische Kommission hat ein Projekt zur Entwicklung einer modellgestützten Fehlermanagement-Software für Verkehrsflugzeuge gestartet.

Langfristig, meinen Forscher, werden die ImmoBots vielleicht allgegenwärtig sein und einige der wichtigsten Infrastruktursysteme steuern. „Wir könnten die unterschiedlichsten Probleme in großem Maßstab lösen, Probleme, die wir bislang niemals bewältigen konnten“, sagt Williams. Wenn er und seine Kollegen Recht haben, lassen sich Infrastruktursysteme mit menschlichen Eigenschaften wie Vorausschau und der Fähigkeit zur kreativen Problemlösung autonom machen, ohne dass dabei das Risiko massiver Störungen durch Software-Probleme steigt. In einer Welt, die immer gefährlicher wirkt, wäre es sicher tröstlich zu wissen, dass ImmoBots auf sich selbst aufpassen können – und auch auf uns. ■

Autonomic Computing: Vorbild Nervensystem

Computer können gewaltig nerven – und mit Anleihen am vegetativen Nervensystem des Menschen will der IT-Konzern IBM genau das ändern.

Im Oktober 2001, auf einer Konferenz für Führungskräfte der IT-Branche, rief Paul Horn nicht weniger als ein „neues Zeitalter der Datenverarbeitung“. Seinen Kollegen gab der Leiter von IBM Research ein Manifest über „Autonomic Computing“ mit auf den Heimweg. Vorbei seien die Zeiten, in denen sich die Industrie mit dem Prinzip „kleiner, schneller, billiger“ zufrieden geben könne, heißt es darin. Die nächste große Herausforderung sei der richtige Umgang mit Komplexität.

Denn während die Kosten für Hardware sinken und sinken, wird es immer aufwendiger – und damit teurer –, sie am Laufen zu halten. Statt nach immer mehr Administratoren zu verlangen, sollen Computer-Systeme sich künftig um sich selber kümmern. Sie sollen in der Lage sein, sich selbst anzupassen und zu optimieren, wenn sich die Anforderungen ändern. Attacken von außen sollen sie aus eigener Kraft abwehren, und wenn das nicht geklappt hat, sich anschließend wenigstens selbst reparieren.

„Es geht darum, die Last von den Menschen zu nehmen“, sagt Alan Ganev, der bei IBM die Autonomic-Computing-Einheit leitet. Computer sollen wie das vegetative Nervensystem

werden, das beim Menschen unauffällig und zuverlässig lebenswichtige Funktionen wie den Herzschlag, den Blutzucker-Spiegel oder die Atemfrequenz regelt.

IBM hat das grundsätzliche Problem treffend beschrieben und eine mitreißende Vision für eine effizientere Zukunft definiert, findet Walter Tichy, Informatik-Professor an der Universität Karlsruhe. „Tichy selbst hat sich davon inspirieren lassen, an seinem Institut Grundlagenforschung auf diesem Gebiet zu betreiben... Es gibt sicher einige Dinge, die kurzfristig verbessert werden können“, sagt Tichy. So hat IBM im März Software für seinen Web-Server WebSphere und sein Datenbankprogramm DB2 vorgestellt, die zumindest einen Teil der „autonomic“-Vision wahr macht: Sie soll etwa auf plötzlich ansteigende Zugriffszahlen mit der automatischen Bereitstellung zusätzlicher Ressourcen reagieren können. Und der Konkurrent Sun Microsystems verspricht noch für dieses Jahr die ersten Ergebnisse seiner „N1“ getauften Initiative. Damit soll ein einziger Administrator 100 statt bisher nur ein Terabyte Speicherplatz verwalten können.

Doch auch wenn schon erste Produkte mit dem „autonomic“-Etikett auf den Markt kommen: „Entscheidende Fortschritte hat es bislang nicht gegeben“, sagt Tichy. Trotzdem darf er keine Zweifel, dass der große Wurf irgendwann kommen wird – vielleicht in fünf Jahren. — Thomas Vasek