

Der digitale Sprachkünstler

Er bezeichnet sich selbst als ein bisschen unkonventionell. Tatsächlich arbeitet **Derek Dreyer** mit Sprachen – aber nicht mit den Sprachen des Alltags, sondern mit Programmiersprachen, weil sie logischer sind. Der gebürtige Amerikaner forscht am **Max-Planck-Institut für Softwaresysteme** in Saarbrücken. Auch wenn die Wissenschaft ein wichtiger Teil seines Lebens ist, gibt es daneben Platz für Leidenschaften wie die Musik. Und dann ist da noch die Sache mit dem Whisky – unkonventionell eben.

TEXT **KLAUS JACOB**

Manche Büros sind so unpersönlich wie der Ausstellungsraum eines Möbelhauses. In anderen kann man lesen wie in einem Buch. Sie verraten viel über den Menschen, der darin arbeitet. So ein Büro hat Derek Dreyer. Eine Stirnseite wird von einer Tafel dominiert, auf der kryptische Gleichungen stehen, mit denen nur Insider etwas anfangen können. Daneben zwei Kinderzeichnungen, die Dereks Tochter gemalt hat. Sie ist vier Jahre alt. „Nein, viereinhalb“, verbessert sich Dreyer. Bei kleinen Kindern zählt jeder Monat.

Die andere Stirnseite füllt ein Regal komplett aus. Anstatt Büchern reiht sich hier ein Whisky an den anderen – aber keine Flaschen, sondern lediglich die Papphüllen. Dreyer ist ein großer Whisky-Fan und kennt fast jede Sorte, aber es muss schon Scotch sein, kein Bourbon.

Das Büro wirkt offen und gemütlich, hell und geräumig. Mittendrin

steht ein bequemes Sofa, auf dem man herrlich lümmeln kann. Die beiden Längsseiten sind völlig verglast. Eine dieser Fensterfronten gibt den Blick frei auf die bewaldeten Berghänge von Saarbrücken, die andere geht auf den Flur. So hat Dreyer zwar ein Büro für sich allein, doch durch die transparenten Wände auch der anderen Räume verbinden sich die Einzelzimmer fast zu einem Großraumbüro.

LOGIK UND MATHEMATIK – DA MACHT IHM KEINER WAS VOR

Und das passt zu Dreyer. Er ist so offen und freundlich wie das Gebäude, scheut vor keiner Frage zurück, lacht gerne und geht auf die Menschen zu. Auf seiner Homepage schreibt er, man könne ihn jetzt „Herr Professor Doktor Derek Dreyer“ nennen, weil er eine Honorarprofessur an der Universität Saarland angetreten habe. Aber lieber solle man ihn einfach „DD“ nennen, wie es seine Tochter Alma tue.

Derek Dreyer ist Informatiker am Max-Planck-Institut für Softwaresysteme, wo er zu Programmiersprachen forscht. Er ist Mitglied der Institutsleitung und Chef der unabhängigen Gruppe „Grundlagen der Programmierung“, wobei ihn zwei Postdocs und sechs Doktoranden unterstützen.

Was Logik und Mathematik angeht, kann ihm niemand etwas vormachen. Gesprochene Sprachen sind dagegen nicht seine Stärke. Obwohl der gebürtige US-Amerikaner schon elf Jahre in Deutschland lebt, muss man sich mit ihm auf Englisch unterhalten. Das liegt nicht zuletzt daran, dass am Institut Englisch die Umgangssprache ist, weil die meisten Forscher aus dem Ausland kommen. Und seine Frau ist ebenfalls US-Amerikanerin, beherrscht Deutsch allerdings deutlich besser. „Es ist sehr peinlich“, sagt Dreyer schmunzelnd auf

Potemkinsche Kollektion: Derek Dreyer sammelt ausgefallene Whiskysorten, im Büro stellt er aber nur die leeren Kartons auf.



Foto: Wolfram Scheible



Ist doch logisch: Dreyer weist nach, dass manche Komponenten etwa der Programmiersprache Rust keine strenge Sicherheitsschleuse benötigen.

Deutsch, mit deutlichem Akzent. Wenn bei ihm zu Hause etwas zu erledigen sei, bei dem man Deutsch sprechen müsse, springe seine Frau ein.

Die deutsche Sprache zu lernen, sei zu mühsam, sagt der Forscher. Dahinter steckt sicher ein wenig Koketterie, denn er beherrscht die Sprache besser, als er behauptet. Immerhin räumt er ein, dass er die korrekten Vokabeln für jedes Lebensmittel kennt, denn gutes Essen bedeute ihm viel, ob japanisch oder thailändisch, französisch oder italienisch.

Auf Dienstreisen sucht er gerne nach einem guten Restaurant und lädt ein paar Freunde und Bekannte ein. „Man kann mir irgendein Lebensmittel geben, und ich kenne das deutsche Wort“, erzählt er. Allerdings fallen ihm das Hören und Verstehen leichter als das Sprechen. Wenn er seiner Tochter etwas auf Deutsch sage, lache sie ihn aus. Denn die Tochter lernt die fremde Sprache kinderleicht in der Kita.

Ganz anders sieht es aus, wenn es um Mathematik geht. Hier fällt Dreyer vieles ohne Anstrengung zu. Er ist ein Überflieger. Schon in der Schule hatte er auf Wunsch der Eltern mehrere Klassen übersprungen, weil er sich gelangweilt hatte. Sein Studium an der Uni begann er mit 13 Jahren anstatt mit 18, wie es üblich wäre. Mit 17 wechselte er bereits an die Carnegie Mellon University, die Topadresse für Informatik. Auch hier war er seinen Kommilitonen um fünf Jahre voraus.

INFORMATIK WAR DREYER NICHT IN DIE WIEGE GELEGT

Im Programm für die PhD-Studenten stockte Dreyers Durchmarsch auf dem Bildungsweg dann jedoch: „Da hatte ich am Anfang eine harte Zeit“, sagt er. Weil er sein wissenschaftliches Projekt in den ersten Jahren ziemlich orientierungslos anging und kaum Fortschritte machte, wäre er im Jahr 2000 beinahe

von der Uni geflogen. Doch er biss sich durch, veröffentlichte bald erste wissenschaftliche Arbeiten und promovierte schließlich 2005.

Geboren wurde Derek Dreyer 1980 in New York City, zog aber schon bald mit seinen Eltern in den Speckgürtel der Metropole, nach Great Neck auf Long Island. Es ist der Teil New Yorks, den der Autor F. Scott Fitzgerald mit seinem Roman *Der große Gatsby* weltbekannt gemacht hat, wie Dreyer gerne erzählt. Er war das Nesthäkchen, viel jünger als seine drei Geschwister, zwei Brüder und eine Schwester.

Sein Vater arbeitete zunächst als Kinderarzt an der New Yorker Universitätsklinik und machte später als Funktionär Karriere. Er wurde Präsident der Amerikanischen Akademie für Kinderheilkunde, des größten Verbandes. „Er ist sehr erfolgreich“, sagt Dreyer über ihn, „und eine große Inspiration für mich.“ Die Mutter blieb zu Hause und kümmerte sich um Kinder und Haus-

» Er ist Grundlagenforscher. Seine Beweisführungen sind allgemeingültig und helfen, die Programmiersprachen der Zukunft zu gestalten.

halt. Die Informatik war Dreyer also nicht in die Wiege gelegt.

Aber schon in der Schule liebte er die Mathematik und wollte dieses Fach unbedingt studieren. Doch die pragmatischen Eltern rieten ihm, sich auch noch für Informatik einzuschreiben, weil er damit leichter einen Job bekomme. So studierte Dreyer zunächst zweigleisig, Mathematik und Informatik, wandte sich aber später ganz der Informatik zu. Inzwischen ist er froh, dass er dem elterlichen Rat gefolgt ist. Denn bei der Mathematik fehlen ihm die Anwendungen. Er mag die Verbindung von Theorie und Praxis, wie er sie nun täglich findet.

Derek Dreyer glänzte nicht nur in den Disziplinen, in denen es um reine Logik geht. Er entwickelte auch musische Qualitäten. Vor allem das Singen macht ihm Spaß. Schon mit 12, 13 Jahren sang er im Chor der New York City Opera, übernahm sogar einige Soloparts. In Chicago – inzwischen war seine Stimme zum Tenor gereift – unterstützte er den Kirchenchor. Und noch heute bedeutet ihm die Musik viel, vor allem Klassik und Jazz. „Meine liebsten Komponisten sind Bach, Britten und Schostakowitsch“, sagt er. „Ich bin stolz darauf, dass ich mir in diesem Jahr die dritte Cello-Suite von Benjamin Britten erschlossen habe. Keine einfache Musik: Ich musste sie etwa 20-mal hören, jetzt finde ich sie aber unglaublich schön.“

Ein Instrument hat Dreyer allerdings nie gelernt. „Das ist wie die deutsche Sprache“, sagt er schmunzelnd, „man muss zu viele langweilige Stücke spielen, ehe man das Instrument beherrscht.“ Vielleicht liegt es ja an sei-

nem Perfektionismus, dass er sich davor scheut, Dinge anzupacken, die ihm nicht auf Anhieb gelingen. Zum Singen ist er in den vergangenen Jahren nicht mehr gekommen, aber er stept gerne, wenn auch nicht bei öffentlichen Auftritten. Um zu zeigen, wie das geht, springt er kurzerhand auf und macht ein paar Schritte. Es sieht ziemlich professionell aus ...

EINE GROSSE CHANCE IN DEUTSCHLAND

Nach dem Studium und der Promotion an der Carnegie Mellon University trat Dreyer zunächst in Chicago eine befristete Stelle am Toyota Technological Institute an. Die Einrichtung hat nichts mit Autos zu tun, sondern betreibt unabhängige Grundlagenforschung. „Sie ist wie Mini-Max-Planck“, sagt Dreyer. Er war Research Assistant Professor, eine Art Postdoc.

Nach drei Jahren, als die Stelle auslief, bewarb er sich international. Denn er hatte sich inzwischen auf Programmiersprachen spezialisiert – ein Fachgebiet, das wenig nachgefragt war. Die Community, die sich damit beschäftigt, ist zwar in den vergangenen Jahren stark gewachsen, aber noch immer überschaubar. Da kam ein Angebot aus Deutschland wie gerufen.

Er musste nicht lange nachdenken, um die Tenure-Track-Stelle am Max-Planck-Institut für Softwaresysteme anzunehmen, da sie ihm bei entsprechendem wissenschaftlichem Erfolg eine langfristige Perspektive bot. Denn Max-Planck hat auch in den Vereinigten Staaten einen hervorragenden Ruf.

Dreyer greift zu seinem Smartphone und zeigt ein weltweites Ranking aller Institute, die sich mit Informatik und Computersprachen beschäftigen. Das Max-Planck-Institut rangiert ganz weit vorne. „Es war eine große Chance“, sagt er, zumal er sich am Aufbau des Instituts beteiligen konnte. Im Jahr 2008 trat er die zunächst befristete Stelle an, dann, fünf Jahre später, bekam er nach einer internationalen Evaluierung eine unbefristete Stelle mit voller wissenschaftlicher Freiheit.

Wenn man ihn fragt, ob der Wechsel von New York und Chicago nach Saarbrücken ein Kulturschock war, gibt er eine höfliche Antwort: Max-Planck sei „great“ und er arbeite viel, da spiele die Umgebung keine so große Rolle. „Wichtig ist für mich, dass ich fantastische Studenten und Kollegen habe, und Saarbrücken ist ein friedlicher Ort, ein guter Platz zum Arbeiten.“ Abwechslung finde er auf Reisen, auch beruflich komme er viel herum, sogar in Weltstädte wie Paris oder London. Bleibt Deutschland also seine zweite Heimat? In naher Zukunft werde er auf jeden Fall in Deutschland bleiben, sagt er.

Wie es danach weitergeht, wisse er noch nicht. Er habe schon andere Optionen erwogen, aber nichts komme an seine Stelle am Max-Planck-Institut heran. Hier könne er forschen, worüber er wolle, habe ein tolles Team und eine hervorragende Ausstattung. Auch die Kombination von Forschung und Lehre findet er in Saarbrücken ideal. „Außerdem“, sagt er, „hat meine Frau Rose Hoberman einen guten Job am Institut, in dem sie unseren Doktoranden bringt, verständliche Papers zu schrei-

ben und packende Vorträge zu halten. Und ihr Büro ist zwei Türen entfernt von meinem – besser könnte es nicht sein!“

Und einen Nachteil, mit dem er sich herumschlagen muss, würde er auch anderswo nicht los: Die Freundschaften, die er und seine Frau eingehen, halten meist nicht lange: Die meisten Freunde kommen aus dem Institut, Studenten oder Postdocs, und sind nach ein paar Jahren wieder fort. Der Wissenschaftsbetrieb verschlägt sie in alle Welt.

Womit sich Dreyer beschäftigt, ist für Laien ebenso unverständlich, wie es die Formeln an seiner Tafel sind. Ironischerweise geht es um Sprachen, die er eigentlich nicht besonders mag. Doch Programmiersprachen fallen in eine andere Kategorie als Deutsch oder Japanisch, denn sie sind absolut logisch strukturiert.

RUST SOLL UNIVERSELL EINSETZBAR UND SICHER SEIN

Derek Dreyer kümmert sich derzeit vor allem um die relativ junge Sprache Rust, mit der sich komplexe Programme schreiben lassen. Für dieses Fünfjahresprojekt namens RustBelt erhielt er vom Europäischen Forschungsrat (ERC) 2016 den Förderpreis Consolidator Grant über zwei Millionen Euro.

Die Programmiersprache Rust entwickelte Graydon Hoare, ein Angestellter der Softwareschmiede Mozilla – bekannt vor allem für den Webbrowser Firefox –, zunächst allein und in seiner Freizeit. Seit 2009 fördert Mozilla die Entwicklung von Rust, und 2015 wurde die Version 1.0 herausgegeben, zusammen mit dem zugehörigen Compiler, der die Befehle in Maschinensprache übersetzt.

Inzwischen wird Rust vielfach verwendet, etwa von Google, Microsoft,

Facebook oder Dropbox. Sie soll die Schwächen herkömmlicher Sprachen beheben. Denn bisher ist eine Sprache entweder sicher, das heißt, sie überprüft automatisch die Arbeit des Programmierers auf formale Fehler. Oder sie ist komfortabel, lässt dem Programmierer also alle Freiheiten – auch die Freiheit, Fehler zu machen.

Aber ein Fehler in einem komplexen Programm, der sich erst bei ersten Testläufen zeigt, ist ausgesprochen lästig. Die Suche kann einen Programmierer zur Verzweiflung bringen. Rust soll den Spagat schaffen, universell einsetzbar und zugleich sicher zu sein. Um zu verstehen, wie das gelingen kann, muss man etwas tiefer in die Materie einsteigen.

Letztlich verfügt jede Programmiersprache über einen Kontrollmechanismus. Allerdings ist diese Instanz unterschiedlich streng. Je schärfer die Kontrollen, desto mehr schränken sie die Freiheit beim Programmieren ein. Denn der elektronische Türsteher wertet manchmal auch etwas als Fehler, was durchaus sinnvoll ist, und zeigt dem Programmierer die rote Karte. Java gehört in die Kategorie mit strengen Kontrollen. Die Sprachen C und C++ sind gewissermaßen die Gegenstücke.

Die verwendeten Kontrollmechanismen sind weit mehr als ein Rechtsschreibprogramm, wie es Autoren verwenden. Ein Beispiel zeigt, wie sie funktionieren: Jede Sprache benutzt Zeiger, die auf bestimmte Daten im Speicher zugreifen. Das klingt recht simpel. Doch komplexe Programme brauchen viele Zeiger und viele Speicherplätze. Da kann der Programmierer leicht den Überblick verlieren, sodass der verwendete Zeiger möglicherweise einen Speicher abrufen, der Unsinn enthält.

Rust verfügt über einige Mechanismen, um das zu verhindern. So hat jeder Zeiger nur eine begrenzte Lebensdauer, das heißt, er ist nur in einem definierten Programmabschnitt aktiv und wird danach, wenn die zugehörigen Daten überholt sind, automatisch ausgeschaltet. Außerdem können niemals zwei Zeiger gleichzeitig Änderungen am selben Speicherplatz vornehmen. Solch eine Dopplung ist eine häufige Fehlerquelle, weil die Daten im Speicher mit jeder Aktion verändert werden können, was bei einem simultanen Zugriff eventuell zu Konfusion führt.

SICHERHEIT OHNE DIGITALEN TÜRSTEHER

Doch die eingebauten Sicherheiten haben einen Nachteil: Manche Aktionen lassen sich nicht ausführen. So ist etwa beim Erstellen komplexer Datenstrukturen der gleichzeitige Zugriff auf denselben Speicherplatz nötig. Um trotzdem universell einsetzbar zu bleiben, enthält die Sprache Rust die Möglichkeit des „Unsafe-Mechanismus“: Der Programmierer kann festlegen, dass bestimmte Programmkomponenten nicht die strenge Sicherheitschleuse passieren müssen. Wer diese Variante nutzt, läuft natürlich Gefahr, dass sich – wie bei C oder C++ – Fehler einschleichen.

An dieser Stelle beginnt die Arbeit von Dreyer und seinem Team. Er weist nach, dass typische „Unsafe-Komponenten“ sicher sind, sodass man sie getrost nutzen kann, auch ohne Türsteher. Er verwendet dabei halbautomatische Beweisassistenten, seine Beweise werden also automatisch vom Computer überprüft. Letztlich beschäftigt sich der Wissenschaftler mit seiner



Musik und Tanz: Neben der Mathematik begeistert sich Dreyer für Musik, vor allem klassische. Er hat früher sogar in der New York City Opera gesungen. Steptanz praktiziert er allerdings nur privat.

favorisierten Kombination aus Mathematik und Informatik.

Allerdings wäre es falsch zu behaupten, Dreyer gehe es lediglich um die junge Programmiersprache Rust. Er ist Grundlagenforscher. Seine Beweisführungen sind allgemeingültig und helfen, die Programmiersprachen der Zukunft zu gestalten. Rust ist für ihn lediglich ein konkreter Anwendungsfall, wobei er sich hier vor allem mit den kompliziertesten „Unsafe-Komponenten“ beschäftigt.

Der ERC-Consolidator-Grant, den er für dieses Projekt erhielt, ist ein Preis, auf den er besonders stolz ist. Ein anderer ist der Robin Milner Young Researcher Award der ACM SIGPLAN, der Vereinigung der Fachverbände für Programmiersprachen; er ist der international renommierteste Preis für junge Wissenschaftler auf dem Gebiet der Programmiersprachen.

Andere Auszeichnungen, die er etwa für herausragende Veröffentlichungen erhielt, tut er mit einer Handbewegung ab. Wie Nippes stehen sie im Regal vor den Whiskykartons. Sie haben weder einen Ehrenplatz, noch sind sie großspurig in Szene gesetzt. Die Reihe der

Schachteln außergewöhnlicher Whisky-Sorten springt jedenfalls viel mehr ins Auge. Neben dem Genuss dieser Spirituosen, besonderem Essen und vor allem der Musik, widmet er sich auch gern guten Weinen und ist dazu sogar einem Weinclub beigetreten. „Das ist mein Weg, mit Deutschen in Kontakt zu kommen“, sagt er lächelnd.

DAS GELOBTE LAND FÜR DEN SCOTCH-ENTHUSIASTEN

In dem Club haben sich ausgewiesene Weinkenner zusammengetan. Alle drei bis vier Wochen treffen sich etwa ein Dutzend Männer und Frauen, alles Deutsche – bis auf Dreyer. Man lädt sich gegenseitig nach Hause ein und kredenzt Wein aus dem eigenen Keller. Die Weinproben arten keineswegs in Besäufnisse aus, die Fachleute spucken die Proben nach dem Verkosten wieder aus. Das sei „part of the deal“, sagt Dreyer.

Der Scotch-Kenner verfügt über die nötige sensible Zunge, um feine Nuancen zu unterscheiden: „Jeder Jahrgang ist völlig anders“, sagt er. Dennoch hat er ein Handicap: Weinkenner benutzen

ein breites Vokabular, um die Geschmacksrichtungen der verschiedenen Lagen und Jahrgänge zu beschreiben. Da kommt der US-Amerikaner nicht mit: „Ich möchte etwas Interessantes sagen, aber mein beschränkter Wortschatz lässt nur Langweiliges zu.“ So hört er meistens zu. Einmal wurde trotzdem er zum Wortführer: Er führte die Runde durch eine Verkostung von vier unkonventionellen Whiskys: Clynelish, Ben Nevis, Springbank und Ledaig – „unkonventionell, ein bisschen wie ich“.

Für den Scotch-Enthusiasten Dreyer ist Deutschland das Gelobte Land. In den USA ist Scotch Whisky nicht nur wesentlich teurer, sondern oft auch gar nicht zu bekommen. Hierzulande stehen die Regale voll davon. Manchmal fährt Dreyer aber auch nach Schottland, um direkt bei den Destillieren zu kaufen. Ob er eine Lieblingssorte hat? Er überlegt eine Weile, dann greift er zu einem Black & White Blend aus den 1960er-Jahren. „Hervorragende Qualität und mit 100 Euro sehr billig“ – ein Schnäppchen. Er habe ihn gestern gekauft, es sei derzeit sein Favorit. Morgen ist es vielleicht ein anderer. ◀