

Sozionik -

Können Soziologen und KI-Forscher von einander lernen?

Andrea Sieber, Werner Dilger

TU Chemnitz, Fakultät für Informatik, Professur KI, 09107 Chemnitz
e-mail: {Andrea.Sieber}/{Werner.Dilger}@informatik.tu-chemnitz.de

Dieser Beitrag soll skizzieren, wie KI-Forscher denken, an Probleme herangehen und diese bearbeiten, wo wir die Unterschiede zwischen der Arbeitsweise der KI-Forscher und der Soziologen sehen und welche Schlußfolgerungen wir daraus hinsichtlich ihrer Zusammenarbeit ziehen.

Den Begriff „Sozionik“ prägten Malsch et al. (96) für das Grenzgebiet zwischen Soziologie und Künstlicher Intelligenz. Es handelt sich dabei um ein in Deutschland in der Entstehung befindliches Forschungsfeld, das die wechselseitigen Annäherungsversuche - von KI-Forschern an soziale Phänomene und von Soziologen an die Möglichkeiten und Grenzen der KI - zur Erforschung künstlicher Sozialität aufgreifen und befördern will. In ihrem ersten pragmatischen Beitrag in der KI-Zeitschrift wollten Malsch, Florian, Jonas und Schulz-Schaeffer der Leserschaft „verdeutlichen, wie Soziologen denken und an Probleme herangehen, wie beide Disziplinen von einander lernen können und unter welchen Bedingungen eine Kooperation unserer ‘Zunft’ mit der ‘Community’ sinnvoll ist.“ (Malsch et al. 96, S. 6)

Wir wollen hier einerseits die soziologische Perspektive, die im Beitrag von Malsch et al. (96) aufgezeigt wurde, um die informatische ergänzen - denn wir gehen davon aus, daß das „von einander lernen“ ernst gemeint ist. Andererseits wollen wir den Blick, der im Titel der Workshopankündigung auf die Disziplinen als solche gerichtet wird, wieder auf die Personen und ihre Arbeitsweise lenken. Sie sind es, die letztendlich voneinander lernen und zusammenarbeiten sollen, denn Theorien und Programme werden (noch) von Personen entworfen und entwickelt. Von ihnen hängt der Erfolg oder das Scheitern dieser wechselseitigen Annäherung ab.

(Er-)Klärungen

Es mag vielleicht verwundern, weshalb ein KI-Beitrag das Forschungsprogramm "Sozionik" aus der Perspektive der Forschungssubjekte und ihren Arbeitsweisen beleuchtet und dessen Bedeutung für ein Gelingen der gemeinsamen Forschungsarbeit hervorhebt. Wir glauben, daß das bereits ein Resultat unserer seit 1995 bestehenden Zusammenarbeit mit den Mitarbeitern der Professur für Industrie- und Techniksoziologie ist. Diese Professur wird von G. Günter Voß geleitet, ein Vertreter der Münchner subjektorientierten Soziologie, die am dortigen SFB 101 und 333 entwickelt wurde. Dabei wird als "subjektorientierte Soziologie" eine Forschungsperspektive bezeichnet, die sich mit den Wechselwirkungen zwischen Subjekt und Struktur auseinandersetzt (vgl. zur Begründung Bolte/Treutner 1983 und zur weiteren

Entwicklung der Perspektive die Beiträge in Voß/Pongratz 97). Es geht darum zu beschreiben, wie Subjekte Strukturen erzeugen, beeinflussen, verändern und wie die Strukturen durch sie wirksam werden.

Verwundern mag vielleicht auch, weshalb wir von „skizzieren“ sprechen - eine Formulierung, die der Geschichte der Informatik als Wissenschaft geschuldet ist. An der Einführung der Informatik als Forschungsgebiet (vgl. dazu ausführlicher Coy 97) waren in der Bundesrepublik Deutschland hauptsächlich Wissenschaftler beteiligt, die ihre Wurzeln in der Mathematik oder den Ingenieurwissenschaften sahen. Erst später gewannen Wissenschaftler an Einfluß, die diese Perspektive als zu eng für die Informatik betrachteten und eine Erweiterung der theoretischen Grundlagen um philosophische, historische, politische und kulturelle Aspekte forderten (vgl. insbesondere den Sammelband von Coy 92). Abgesehen von der Skandinavischen Schule (insbesondere Nygaard 86, Naur 92) und jenen, die in deren Tradition stehen (vgl. z. B. die Beiträge in Floyd et al. 92), gibt es innerhalb der Disziplin kaum Arbeiten, die die Tätigkeit der Informatiker als solche methodisch fundiert reflektieren.

Was könnte der Grund dafür sein? Eine Reflexion dieser Tätigkeit - wenn sie methodisch fundiert sein soll - kommt nicht an den Methoden empirischer Sozialforschung vorbei. Diese Methoden fanden in der Vergangenheit aber hauptsächlich Anwendung in der Soziologie oder der Psychologie. Deshalb finden sich dort eine Reihe von Arbeiten zur Beschreibung von Programmierarbeit (vgl. neben den hier im folgenden zitierten Arbeiten insbesondere Soloway/Iyengar 86, Olson/Sheppard/Soloway 87, Hoc 90, Guindon 90, Davis 90, Rose 90, Weltz/Ortmann 92, Frese/Brodbeck 94, Grudin 97). Allerdings sind die Ergebnisse dieser Arbeiten nicht ohne weiteres für die Informatik verständlich und brauchbar, da die Untersuchungen in einer anderen Perspektive und mit von der jeweiligen Disziplin bestimmten Zielen erfolgten.

Das bedeutet aber auch - und jetzt kommen wir auf unsere Absicht, die Arbeitsweise von KI-Forschern zu „skizzieren“ zurück - daß der wissenschaftliche Erkenntnisstand hinsichtlich der Arbeits- und Denkweise von Informatikern eher punktuell ist, als daß wir in der Informatik bereits ausgearbeiteten Theorien dazu gegenüberstehen.

Doch wieso läßt sich von der Arbeits- und Denkweise von Informatikern auf jene der KI-Forscher schließen? Wir gehen davon aus, daß das Programmieren ein wichtiger Bestandteil der Tätigkeit beider Personengruppen ist. Wenn wir die mit dem Programmieren in Zusammenhang stehenden Mechanismen, Denkweisen und Vermittlungsleistungen der Softwareentwickler darstellen können, gewinnen wir Einblick in einen wichtigen Aspekt des Tuns sowohl bei Informatikern allgemein auch als bei KI-Forschern speziell. Doch bevor wir diese Wechselwirkungen versuchen zu beschreiben, werden wir noch einmal zusammenfassend aufzeigen, inwieweit die VKI aus der Kritik an der KI hervorging, um zu verdeutlichen, welche Vorstellungen und Theorien dieser Disziplin zugrundeliegen. In der anfänglichen Kritik an der KI von Weizenbaum (78) sind dann auch erste Ansätze zur Charakterisierung der Tätigkeit des Programmierens - basierend auf der Reflexion eigener Erfahrungen des Autors - zu verorten. Auf KI-Forscher übertragbare Merkmale dieser Tätigkeit werden darauf aufbauend erarbeitet. In einem dritten Schritt wollen wir anhand unserer eigenen Erfahrungen hinsichtlich der Zusammenarbeit zwischen KI-Forschern und Industriesoziologen und der Selbstbeschreibung der soziologischen Denkweise durch Malsch et al. (96) prinzipielle Unterschiede der Arbeitsweisen zwischen beiden Disziplinen herausarbeiten. Am Schluß skizzieren wir resümierend, wie in der Sozionik ein konstruktives Arbeiten möglich sein könnte.

Die Entstehung der VKI als Reaktion auf die Kritik an der KI

Bereits 1972 formulierte Dreyfus vier Annahmen (Dreyfus 89, S. 103ff.), die den Arbeiten von KI-Forschern zugrunde liegen: Sie sehen bei der Nachbildung intelligenten Verhaltens einen engen Zusammenhang zwischen den Vorgängen im Gehirn und den Vorgängen im Digitalcomputer. D. h., sie gehen davon aus, daß ein Gehirn mit seinen Neuronen genauso funktioniert wie eine Schaltung, die aus Transistoren besteht (biologische Annahme) und versuchen, mit dem Computer ein Modell des Denkens zu entwerfen (psychologische Annahme). Sie gehen zudem davon aus, daß bewußtes Wissen für die Verarbeitung im Computer formalisiert werden kann, d. h. sie können Regeln angeben, wie eine Person zu diesem Wissen gelangt bzw. woraus es resultiert (erkenntnistheoretische Annahme). Wissen als solches besteht demzufolge aus kontextunabhängigen und festgelegten Elementen, die logisch miteinander verknüpft sind (ontologische Annahme).

Mittlerweile ist die Anzahl der Vertreter der 'starken' KI zurückgegangen. Eine sogenannte 'schwache' KI hat sich etabliert, die nicht den Anspruch hat, menschliches Verhalten oder menschliche Intelligenz im Computer als solches abzubilden. Ihr geht es nicht um eine Vorstellung von „allem“ Wissen oder um die Realisierung menschlichen Denkens im formal-logischen Sinne. Sie will bestimmte Aspekte menschlichen Verhaltens oder menschlicher Intelligenz simulieren, um so intelligente, adaptive Werkzeuge für den Gebrauch durch Menschen zu schaffen. Zur regelbasierten Vorgehensweise sind weitere Modelle und Mechanismen hinzugekommen, die sich an Modellen anderer Wissenschaften orientieren. Fuzzy-Logik, evolutionäre Algorithmen, Lernen aus Beispielen oder Neuronale Netze - damit soll bei der Verarbeitung von ankommenden Informationen die explizite Digitalisierung und Formalisierung des Wissens umgangen werden. Für Dreyfus Grund genug, seine Kritik von 1972 zu erneuern (vgl. Dreyfus 89). Er ist der Meinung, daß mit dem Verzicht auf eine explizite Formulierung der Regeln - damit, daß ich dem Computer das Finden der Regeln, wie z.B. eine Beispielmenge konstruiert ist, überlasse - das Grundproblem, daß Wissen als Wissen um Regeln verstanden wird, nicht beseitigt wurde.

Mit der Einführung mehrerer künstlich intelligenter Einheiten, deren Struktur und Verhalten Gegenstand der VKI-Forschung ist, versucht unserer Meinung nach die KI-Forschung, Wissen im Gegensatz zur bisherigen Annahmen als verteilt zwischen verschiedenen Akteuren und abhängig von deren Aufgaben zu verstehen und programmtechnisch umzusetzen. Sie gesteht einzelnen Akteuren Spezialwissen und für die Lösung ihrer Probleme spezifische Vorgehensweisen zu.

Über Arbeiten von Maturana und Varela (80) zur biologischen Erkenntnis sowie von v. Glaserfeld und v. Foerster u. a. (Watzlawick 81) zur Konstruktion von Wirklichkeit fanden die Ideen zur körpergebundenen und kontextabhängigen Erkenntnis ihren Weg in die KI. Die maschinelle Konstruktion menschenähnlichen Verhaltens scheint nun nur noch unabhängig vom Computer möglich. Autonome Systeme, die mittels Sensoren raum- und zeitabhängige Informationen aus ihrer Umgebung aufnehmen können, sollen ihre eigenen Wissenskategorien konstruieren (vgl. Wellner 98).

Daß die Realisierungen solcher Vorhaben nach wie vor „aus der Perspektive des methodologischen Individualismus entworfen werden und 'Gesellschaft' als eine Ansammlung von rationalistischen Einzelakteuren aufgefaßt wird.“ (Malsch et al. 96, S. 9) ist unserer Meinung nach nicht verwunderlich, wenn man die Besonderheiten der Tätigkeit des Programmierens, wie sie auch von KI-Forschern ausgeübt wird, zur Kenntnis nimmt.

Was bedeutet es zu programmieren?

Schon 1976 kritisierte Joseph Weizenbaum die Dominanz der instrumentellen Vernunft bei der Lösung von Problemen mit Hilfe eines Computers. Er legte unter Bezug auf Maslow dar, welchen Einfluß die Programmierung von Computern mit Hilfe von Programmiersprachen auf die Weltsicht der Programmierer habe. „Der Psychologe Abraham Maslow hat einmal gesagt, für jemanden, der nur einen Hammer und sonst nichts habe, sehe die ganze Welt wie ein Nagel aus. Auch eine Sprache ist ein Werkzeug, und auch sie bestimmt in der Tat die Form, die die Welt für den annimmt, der sich ihrer bedient.“ (Weizenbaum 78, S. 142).

Dabei sind Programmiersprachen nicht nur „angenehme Kürzel, mit denen sich die Erscheinungen ihres Fachgebietes beschreiben ließen“. Sie werden „immer umfassendere Theoriegebäude“, die nicht nur der Beschreibung von Welt dienen, sondern gleichzeitig zu „Determinanten der Weltsicht dieser Wissenschaftler“ werden (Weizenbaum 78, S. 143). Das wird auch daran deutlich, daß Programmiersprachen als „Paradigmen“ im Sinne von Kuhn (70) bezeichnet werden. Die von Nygaard (86) angestrebte gleichberechtigte Integration der verschiedenen Ansätze (imperativ, objektorientiert, deklarativ etc.) in den Systembeschreibungen und Programmiersprachen - um so den verschiedenen Sichten auf ein Problem gerecht zu werden - hat sich zumindest in dem von Sieber untersuchten Bereich der Softwareentwicklung immer noch nicht etabliert. In der empirischen Untersuchung zur Praxis des Programmierens (vgl. Sieber in Vorb.) - es handelt sich um Intensivfallstudien in sechs kleinen und mittleren Unternehmen - deutet sich eher an, daß mit jeder neuen Generation an Informatik-Studenten eine neue Programmiersprache zum „Paradigma“ erhoben wird, die dann automatisch als besonders geeignet für die Lösung der anstehenden Probleme in Erscheinung tritt. Weil aber die anderen Entwickler-Generationen ihre „Weltsicht“ nicht einfach beiseite legen und auch nicht mehrere nebeneinander haben können, kommt es dadurch permanent zu Konflikten im Arbeitsteam, die jene Personen mit mehr Machtmitteln für sich entscheiden.

Für Weizenbaum ist das einzige Werkzeug der Programmierer die Programmiersprache. Damit schaffen sie auf der realen Maschine, die lediglich die Zustände 0 und 1 kennt, virtuelle Maschinen, die ihren Vorstellungen unterliegen, die ihre Gesetze und Regeln befolgen und die in keinem Zusammenhang mit der realen Welt stehen müssen (sie müssen z. B. keinen physikalischen Gesetzen folgen). Sie können ihrer Kreativität freien Raum lassen, sie können ihre eigene Welt erschaffen, in der nur sie die Herrscher sind. Daraus entsteht für Weizenbaum eine „Omnipotenz“, eine Vorstellung von Macht, die den Programmierer korrumpiert und zu zwanghaftem Handeln führt (Weizenbaum 76, S.160).

Heutige Softwareentwickler haben nicht nur mächtige Programmiersprachen, um sich auf dem Computer ihre eigene Welt zu schaffen. Mit Hilfe des Internet können sie - ohne ihren Platz vor dem Computer zu verlassen - genauso intensive soziale Kontakte pflegen und Informationen aller Art erhalten. Deshalb charakterisiert Schelhowe (97, S. 10) den Computer in seiner heutigen Benutzung durch Informatiker als „instrumentales Medium“: Es ist Werkzeug und Informationen aus aller Welt vermittelndes Medium zugleich.

Welchen Einfluß hat dieser Umgang mit dem Computer auf das Denken und Handeln der Personen, auf ihre Weltsicht und ihren Weltbezug? Schachtner (93) versucht in ihrer empirischen Studie aus psycho-analytischer Sicht den Bezug von Softwareentwicklern zu ihrer Tätigkeit und zur Welt nachzuzeichnen. Sie kommt zu dem Schluß, daß ihre Arbeitsweise stark geprägt ist durch das Aushalten von Ambivalenzen. Einerseits erfordert Programmierung

formal-logisches Denken. Das, was Softwareentwickler von ihrer Welt verstehen, müssen sie zerlegen, formalisieren und verregeln, um der Logik der Maschine gerecht zu werden. Damit einher geht eine Betonung kognitiver Denkprozesse, eine Betonung der Arbeit mit dem Kopf. Andererseits bemerken Softwareentwickler diese Betonung und versuchen sie zu umgehen. Das erfolgt mehr oder weniger bewußt, indem sie emotionale und körperliche Empfindungen verdrängen, diese mit Hilfe von extremen körperlichen Belastungen kompensieren oder ihrer Arbeit gleichzeitig begeistert und distanziert gegenüberstehen.

Neben dieser je spezifischen Problemsicht entwickeln Programmierer zudem subjektive Arbeitsstile hinsichtlich ihrer Organisation von Arbeitsschritten, der Darstellung von Zwischenergebnissen, des Tests und der Fehlersuche sowie der Nutzung von Softwaretools (vgl. Strübing 93, S. 243ff.). Nach Strübing deutet sich in seinem empirischen Material zudem an, daß Entwickler gehäuft in ihrem Gegenstandsbereich „auf Neues“ orientiert sind, eher abgeneigt sind, den technischen Sachbezug durch eine Karriere zu verlieren, die Bedeutung des Kollektivs für ihre Arbeit hervorheben und in ihren Aussagen betont sachorientiert vorgehen.

Nach den Untersuchungen von Sieber (in Vorb.) variiert die Bedeutung des Kollektivs in Abhängigkeit vom zu lösenden Problem. Handelt es sich um Schwierigkeiten in der Auseinandersetzung mit der Maschine, dann unterstützen sie sich in der Regel immer. Geht es um die inhaltliche Strukturierung einer Arbeitsaufgabe, dann können sie nur konstruktiv zusammenarbeiten, wenn ihre Vorstellungen hinsichtlich der Art und Weise, wie Probleme zu formalisieren sind (ihre „Weltsicht“), übereinstimmen.

Bezüglich der Vorgehensweise bei der Entwicklung von Softwareprodukten deutet sich bei Sieber (in Vorb.) an, daß sie zuerst ihre zentralen Vorstellungen vom Problem und seiner informatischen Lösung umsetzen (erster Prototyp). Dann antizipieren sie aufgrund ihrer bisherigen Erfahrungen mit den Personen im Anwendungsbereich ihrer Software, welche Ansprüche diese an ihre Software stellen könnten. Der konkrete Bezug zur ‘Außenwelt’ erfolgt über Kontaktpersonen, die ihre Problemsicht in bestimmten Punkten teilen und die bezüglich des Umganges mit Technik bereits einige Erfahrungen haben. Und erst zu diesem Zeitpunkt beginnen die Entwickler, das Software-Produkt tatsächlich in den Arbeitskontext der Anwender einzupassen.

Ausgehend von unserer Annahme, daß Programmieren ein wichtiger Bestandteil der Tätigkeit von KI-Forschern ist, wollen wir an dieser Stelle die für unsere weitere Argumentation wichtigen Zusammenhänge noch einmal kurz darstellen:

- Die Werkzeuge, mit denen KI-Forscher ihre Anwendungen konstruieren, können von ihnen nach ihren Regeln gestaltet werden. Diese sind dabei gleichzeitig Determinanten ihrer „Weltsicht“.
- Die Gestaltung der Werkzeuge nach ihren Regeln führt zu Machtvorstellungen hinsichtlich der Kontrollierbarkeit ihrer Produkte und zur Abhängigkeit vom Computer, der ihnen das ermöglicht.
- Unterschiedliche „Weltsichten“ in einem Arbeitsteam führen zu Konflikten zwischen den jeweiligen Personen, die in der Praxis oft mit Machtmitteln unterdrückt werden.
- KI-Forscher müssen große Ambivalenzen zwischen dem Denken und dem Fühlen aushalten, wobei aushalten auch verdrängen oder kompensieren heißen kann.
- KI-Forscher sind gehäuft „auf Neues“ orientiert: Sowohl im Bereich der Hardware (neue Technik) als auch im Bereich der Software (neue Programmierparadigmen).
- Bei Entwicklungen setzen KI-Forscher zuerst ihre eigenen Ideen nach ihrem Verständnis um und entwickeln diese dann aufgrund der Auseinandersetzung mit andern Personen weiter.

Die Arbeitsweise von Soziologen im Vergleich

Leider sind die folgenden Aussagen über das, was Soziologen tun, alles andere als wissenschaftlich. Sie spiegeln ein Stück weit das wieder, wie wir Soziologen erfahren haben und wo es bei uns oder unseren Kollegen immer wieder Verständigungsschwierigkeiten gibt. Wir wollen diese Dinge trotzdem thematisieren, weil wir glauben, daß das Forschungsprogramm „Sozionik“ nur Erfolg haben wird, wenn KI-Forscher und Soziologen um eventuelle Unterschiede wissen.

Malsch et al. (96, S. 8) nennen vier inhaltliche Punkte, aufgrund dessen Soziologen dem Unternehmen „Sozionik“ skeptisch gegenüber stehen:

- soziale Tatsachen lassen sich nur natürlichsprachlich formulieren,
- Soziales läßt sich nicht in für die Programmierung brauchbare Modelle passen,
- formale Modelle geben die Reflexivität sozialer Verhältnisse nur ungenügend wieder,
- die Komplexität der Computerdarstellung bleibt hinter der Komplexität einer sozialen Welt weit zurück.

Wir glauben, daß sich beide Personengruppen auch skeptisch aufgrund ihrer unterschiedlichen Arbeitsweise gegenüberstehen:

Soziologen befragen mit Vorliebe Leute und beschreiben Phänome der Wirklichkeit. Sie suchen nach Erklärungen für das, was sie beobachten, aber sie sind Beobachter aus der Distanz. Sie wollen nicht eingreifen und schon gar nicht, wenn es um Technik geht. Denn die Welt ist komplex und die Zusammenhänge darin so vielfältig, daß jedes Handeln unüberschaubare Folgen haben kann.

KI-Forscher bilden mit Vorliebe Phänome der realen Welt im Computer nach. Dabei beginnen sie mit einfachen Beispielen, treffen für Ausnahmen ein paar Festlegungen und schon ist das Problem für die Bearbeitung mit dem Computer brauchbar. Sie wissen, daß ihnen sowieso immer wieder Verbesserungen für ihre Programme einfallen, daß es immer wieder neue Entwicklungen geben wird, die diese oder jene Schwierigkeit beheben werden. Das hält sie jedoch nicht davon ab, an „ihr“ Programm zu glauben, sich damit zu identifizieren, es weiterzuentwickeln und zu verteidigen.

Soziologen haben völlig andere Lesegewohnheiten als KI-Forscher. Sie lesen schwer verständliche, weit ausschweifende, mit Fremdwörtern gespickte Texte, an denen scheinbar die Länge des Literaturverzeichnisses das entscheidende Kriterium ist. Den Autoren geht es darum zu zeigen, was sie alles wissen und nicht darum, daß die Leser sie verstehen. Solche Texte mögen KI-Forscher überhaupt nicht. Wenn sie schon Text lesen (und keine Formeln und Grafiken anschauen), dann am liebsten Science Fiction: Das ist kurzweilig, ein bißchen wissenschaftlich und vor allem anregend, interessant, informativ, eingänglich und spannend.

Soziologen messen Sprache einen völlig anderen Wert bei als KI-Forscher. Sie halten gern Vorträge, denen man kaum folgen kann, weil die Sätze so lang sind. Sie können stundenlang über die Bedeutung eines Wortes oder einer Aussage diskutieren und finden das auch noch interessant! KI-Forscher lassen da lieber Tatsachen in Form von Computerprogrammen für sich sprechen lassen. Oder sie stellen einfach provokante Thesen auf. Dann haben die anderen das Problem, Gegenargumente zu bringen!

Das bereits vorhandene Wissen in der Soziologie ist schier unendlich (Institutionalisierung 1887 in Bordeaux). Wer sich mit Soziologie beschäftigt, der beschäftigt sich zuerst einmal mit der Aufarbeitung, Interpretation und Ergänzung dessen, was die verschiedensten Leute in diesem gesamten Zeitraum gesagt, was andere darauf erwidert haben und wie es mittlerweile konkretisiert wurde. KI-Forscher interessiert Geschichte nicht. Ihr Fachgebiet wurde 1960 in den USA institutionalisiert. Was die Leute damals dachten ist heute sowieso nicht mehr relevant. Und wenn es morgen wieder etwas Neues gibt, dann probieren sie das aus!

Resümee

Diese - wissenschaftlich nicht fundierten Behauptungen - offenbaren eines sofort: Soziologen arbeiten anders als KI-Forscher, lesen anders, messen Sprache einen anderen Wert bei, gehen mit Geschichte anders um als sie:

- Während die einen nur beschreiben und erklären wollen, greifen die anderen konkret ein.
- Während für die einen Inhalte wichtig sind, wollen die anderen eine möglichst übersichtliche Struktur.
- Während für die einen Geschichte allgegenwärtig ist, empfinden sie die anderen als Ballast.
- Während die einen mit Vorliebe Realitäten reflektieren wollen die anderen eigene Realitäten schaffen.

Auf den ersten Blick scheint es nahezu unmöglich, daß so unterschiedliche Personengruppen von einander lernen können. Es verbindet sie kaum mehr als die Tatsache, daß sie sich über den gleichen Gegenstand Gedanken machen und daß sie sich Wissenschaftler nennen. Was bedeutet das für die Umsetzung soziologischer Theorien, mit dem Ziel, die VKI-Forschung voranzubringen? Muß erst die 'virtuelle Maschine' geändert werden, damit andere Modelle umsetzbar sind? Oder müssen VKI-Forscher und Soziologen, die im Bereich der Sozionik zusammenarbeiten, erst neue Arbeitsroutinen entwickeln, um überhaupt konstruktiv miteinander arbeiten zu können?

Vielleicht kommen wir dem Forschungsgegenstand Sozionik und einander näher, wenn wir herausfinden, was der Hintergrund dieser beabsichtigten Zusammenarbeit ist: Warum wollen Soziologen den VKI-lern helfen, bessere Modelle zu entwickeln? Warum wollen sie das, von dem sie überzeugt sind, daß es nicht adäquat simulierbar und formal-logisch darstellbar ist, trotzdem computerbasiert in Simulationsmodellen umsetzen? Woher kommt ihr Glaube - den mancher Informatiker gerade zu verlieren scheint und sich im Gegenzug für eine Selbstbeschränkung hinsichtlich des Computereinsatzes zu erwärmen beginnt - daß sich mit Computern eine künstliche Sozialität realisieren läßt? Wollen sie den Informatikern helfen, eine soziale Technik zu konstruieren, an der auch die letzten Soziologen oder Technikkritiker nicht mehr vorbeikommen?

Wir stellen diese Fragen in der Hoffnung, daß wir ins Gespräch kommen und beginnen können, von einander zu lernen. Unserer Meinung nach ist eine konstruktive inhaltliche Zusammenarbeit nur möglich, wenn die Unterschiede und Gemeinsamkeiten in den Vorstellungen der Personen aus den so unterschiedlichen Disziplinen expliziert und reflektiert werden. Nur so können wir wenigstens zu Verständnis für die Problemsicht des jeweils Anderen gelangen und beginnen, uns damit auseinanderzusetzen - im ganz konkreten Miteinander, indem wir wechselseitig erleben, wie wir arbeiten, denn nur so erhalten KI-Forscher überhaupt einen Zugang zur Welt soziologischer Theorien und können ein Verständnis davon entwickeln, das wiederum zu adäquateren Programmen führt.

Nach unseren Erfahrungen gehen damit aber auch Verunsicherungen einher, Vorstellungen vom Leben verändern sich, die Neuorientierungen nötig machen. Auch dafür sollte Platz beim von einander Lernen sein. Raum und Zeit, veränderte Vorstellungen über das Leben und Zusammenleben zu reflektieren, Orientierungsverluste zu artikulieren und Konflikte zu thematisieren.

Wir müssen uns unserer Meinung nach auf einen langwierigen Prozeß einstellen, der wohl nur zu einem Bruchteil mit informatischer Logik zu erfassen ist - aber er verspricht neben Konflikten, Schwierigkeiten und unerwarteten Hindernissen auch viel Neues, Spannendes und Interessantes auf Gebieten, die sowohl der Informatik als auch der Soziologie bisher nur wenig zugänglich waren.

Literatur:

- Bolte, Karl Martin; Treutner, Erhard (Hrsg.): Subjektorientierte Arbeits- und Berufssoziologie. Frankfurt ; New York : Campus, 1983.
- Brodbeck, Felix C.; Frese, Michael (Hrsg.): Produktivität und Qualität in Software-Projekten. Psychologische Analyse und Optimierung von Arbeitsprozessen in der Software-Entwicklung. München : Oldenbourg, 1994.
- Coy, Wolfgang et al. (Hrsg.): Sichtweisen der Informatik. Braunschweig ; Wiesbaden : Vieweg, 1992.
- Coy, Wolfgang: Defining Discipline. In : Freska, Ch.; Jantzen, M.; Valk, R. (Hrsg.): Foundation of ComputerScience: Potential-Theory-Cognition. Berlin ; Heidelberg ; New York et al. : Springer, 1997.
- Davis, Simon P.: The Nature and Development of Programming Plans. In : International Man-Machine Studies, 32, 1990, S. 461-481
- Dreyfus, Hubert L.: Was Computer nicht können. Die Grenzen künstlicher Intelligenz. Frankfurt am Main : Athenäum, 1989. (amerik. Original: What Computers can't do - The Limits of Artificial Intelligence. New York : Harper & Row, 1972)
- Dreyfus, Hubert L.: What Computers still can't do. A Critique of Artificial Reason. Cambridge ; London : The MIT Press, 1992. (Rev. ed. of: What computers can't do, 1979)
- Floyd, Christiane; Züllinghoven, Heinz; Budde, Reinhard; Keil-Slawik, Reinhard (Hrsg.) Software Development and Reality Construction. Berlin et al. : Springer, 1992.
- Guindon, Raymonde: Designing the Design Process: Exploiting Opportunistic Thoughts. In : Human-Computer-Interaction, 5, 1990, S. 305-344
- Grudin, Jonathan; Palen, Lesia: Emerging Groupware Successes in Major Corporations: Studies of Adoption and Adaptation. In : Mambrey, P. et al. (Hrsg.): Rechnergestützte Kooperation in Verwaltungen und großen Unternehmen. Universität -GHS- Essen, 1997, S. 235-245 [Tagungsband zum Workshop]
- Hoc, J.-M. et al. (Hrsg.): Psychology of Programming. London, 1990, S. 205-222.
- Kuhn, Thomas S.: The Structure of Scientific Revolution. 2nd edition, Chicago : The University of Chicago Press, 1970.
- Malsch, Thomas; Florian, Michael; Jonas, Michael; Schulz-Schaeffer, Ingo: Sozionik. Expeditionen ins Grenzgebiet zwischen Soziologie und Künstlicher Intelligenz. In : KI, 2/1996, S. 6-12
- Maturana, H. R.; Varela, F. J.: Autopoiesis and Cognition. Dordrecht : D. Reidel Publishing Company, 1980.
- Naur, Peter: Computing: A Human Activity. New York et al : ACM-Press, 1992.
- Nygaard, Kristen: Program Development as a Social Activity. In : Kugler, H. J. (Hrsg.) Information Processing '86: Proceedings of the IFIP 10th World Computer Congress, Dublin, September 1-5, Amsterdam : North-Holland, 1986, S. 189- 198
- Olson, G. M.; Sheppard, S.; Soloway, E. (Hrsg.): Empirical Studies of Programmers. Second Workshop. New Jersey : Norwood, 1987.
- Rose, H. (Hrsg.): Programmieren in der Werkstatt. Frankfurt ; New York : Campus, 1990.
- Schachtner, Christel: Geistmaschine. Faszination und Provokation am Computer. Frankfurt am Main : Suhrkamp, 1993.

- Schelhowe, Heidi: Das Medium aus der Maschine. Zur Metamorphose des Computers. Frankfurt am Main ; New York : Campus Verlag, 1997.
- Sieber, Andrea: Softwareentwicklung in der Praxis. TU Chemnitz, Fakultät für Informatik, Dissertation. In Vorbereitung.
- Soloway, E.; Iyengar, S. (Hrsg.): Empirical Studies of Programmers. First Workshop. New Jersey : Norwood, 1986.
- Voß, Günter G.; Pongratz, Hans J. (Hrsg.): Subjektorientierte Soziologie. Opladen : Leske und Budrich, 1997.
- Watzlawick, Paul (Hrsg.): Die erfundene Wirklichkeit. München ; Zürich : Piper, 1981.
- Weltz, Friedrich; Ortmann, Rolf: Das Softwareprojekt - Projektmanagement in der Praxis. Frankfurt, New York : Campus-Verlag, 1992.
- Weizenbaum, Joseph: Die Macht der Computer und die Ohnmacht der Vernunft. Frankfurt am Main : Suhrkamp, 1978. (amerik. Original: Computer Power and Human Reason. From Judgement to Calculation. San Francisco : W.H. Freeman and Company, 1976.)
- Wellner, Jörg: Machine Epistemology for Artificial Life. Vortragsmanuskript 1998.