

Mein Weg zu einer ökologischen Orientierung in der Informatik

Im Winter 1976/77 verbrachte ich ein Forschungssemester an der Purdue University in den USA. Meine Frau und ich wollten etwas gemeinsam tun – außer vier Kinder bändigen – und gerieten in einen Literatur-Kurs „Literature of Ecology“, „Das Umweltbewußtsein in der Schönen Literatur“. Wir lasen und diskutierten wunderbare Bücher – von „Walden“ von Henry David Thoreau, dem Klassiker der amerikanischen Umweltbewegung vor 150 Jahren, über „The Wilderness World of John Muir“ und „A Sand County Almanac“ von Aldo Leopold bis zu „Pilgrim at Tinker Creek“ von Annie Dillard, einer zeitgemäßen Nachfolgerin von Thoreau. Wie diese gingen wir in die Natur „to see what there is to see“ und schrieben Tagebuch. Es dauerte eine Weile, bis wir in dem traurigen Waldstück im harschen Winter des Mittleren Westens etwas sahen. Aber dann sahen wir immer mehr und schrieben immer freier. Wir konnten nicht wie sie im Wald heimisch werden; stattdessen richteten wir uns im Kurs gemeinsam wohnlich ein. Unser ökologisches Bewußtsein erwachte.

„Du mußt es in Deine Technische Universität hineinragen“, predigte mir der Dozent. „Wir brauchen ökologisches Bewußtsein in Wissenschaft und Technik.“ Dabei half mir E.F. Schumacher, der sich sein Leben lang für die Gedanken Thoreaus im wirtschaftlichen und technischen Bereich eingesetzt hat und für mich der Vater der Idee der „nachhaltigen Entwicklung“ ist.¹ Nach dem zweiten Purdue-Winter 1979/80 war ich soweit. Ich hielt an der TU Berlin Seminare mit Informatik-Studenten zur „Literatur kleiner Systeme“ ab; das Wort ‚klein‘ borgte ich von E.F. Schumachers „Small is Beautiful“, die Systeme kamen aus der Informatik. Wir lasen zunächst dieselben Bücher, aber mit informatischem Blick. Die Welt ist unermesslich, aber die Umwelten, in denen wir leben, sind klein, überschaubar. Oder? Die Uni oder Rechnersysteme überschaubar? Sie sind es nicht von selbst. Dauernd finden wir uns bestimmt von gesellschaftlichen Bedingungen, sprachlichen Gegebenheiten, ethischen Normen, die universell gelten. Aber wir können nicht leben, ohne uns in den großen Rahmen an allen Ecken kleine vertraute Umgebungen einzurichten. Die geringe Zahl tut's freilich nicht. Jede Zweierbeziehung kann emotional erstarren, sprachlich vertrocknen, materiell ersticken; und wir fühlen uns so verloren wie ein Kind im Kaufhaus. „Der kleine Dirk erwartet seine Mami an der Umtauschkasse.“ Die Qualität kleiner Systeme ist eine menschliche und nicht formal faßbar; sie liegt in den Spielräumen, die wir anderen und uns materiell, emotional und sprachlich lassen; zahlenmäßige Grenzen ergeben sich daraus, nicht umgekehrt. Kleine Systeme sind eine Aufgabe, keine Einteilung der Welt.

In der Informatik sah ich es zuerst an Lehrveranstaltungen. Die meisten Studenten lernen am besten in kleinen Gruppen, wenn sie gemeinsam Probleme knacken und Fragen diskutieren können. Vorlesungen sind gut, um Anreize zu geben, Fragen aufzuwerfen, Verknüpfungen herzustellen – wenn die Dozenten das wollen und nicht nur Stoff rüberschieben. Allein zu Hause nacharbeiten ist nötig, um das Angefangene sich wirklich anzueignen und in Ruhe zu überdenken – nicht um Stoff zu pauken. Beide, die große Vorlesung wie der einsame Schreibtisch, bekommen ihren Sinn erst aus der gemeinsamen Arbeit, sie sind notwendige Hilfen, nicht Selbstzweck. So kam ich von Seminaren über kleine Systeme zu Lehrveranstaltungen im Sinne und zur Förderung kleiner Systeme. Meine Kollegin Christiane Floyd und ich erarbeiteten uns in Seminaren über Hubert Dreyfus, Gregory Bateson und Niklas Luhmann ein kritisches Verständnis der Informatik, das sich in der Lehr- und Forschungsarbeit unserer Gruppen in Softwaretechnik bzw. Theoretischer Informatik konkretisierte.²

Was sollen Geistes- und Sozialwissenschaften in der Informatik? konnten Kollegen sagen; in der Informatik geht es um Rechner und Programme, die sind nicht „klein“ zu kriegen. Ich benutze ‚klein‘ und ‚groß‘ als soziale Begriffe, anwendbar nicht auf Maschinen und ihre formalen Bedienungsanweisungen, sondern auf Gruppen von Menschen, die solche herstellen, benutzen oder wegschmeißen; das ist richtig. Aber in der Informatik geht es nicht nur um Rechner und Programme, sondern darum, was wir mit ihnen tun. Wir müssen verstehen, was wir mit ihnen bewirken, positiv wie negativ, bei anderen und bei uns selbst. Wenn wir bei diesem gemeinsamen Tun uns um kleine Systeme bemühen, programmieren und konstruieren wir „auf kleine Weise“. Wenn wir nur auf die Objekte sehen, „auf große Weise“ mit ihnen umgehen, erzeugen wir Angst und Abhängigkeit. Aus der Beschäftigung mit Natur und Literatur kam ich in der Informatik zu einer Theorie der Formalisierung.³

Ende 1988 rief Wolfgang Coy einen Arbeitskreis „Theorie der Informatik“ ins Leben, in dem ich mitarbeitete. Der Arbeitskreis war im Fachbereich „Informatik und Gesellschaft“ der Gesellschaft für Informatik angesiedelt, deren Sprecher Coy damals wurde, und wurde vom BMFT als Diskursprojekt gefördert.⁴ Der Titel war programmatisch gemeint: Die Mathematik hat in der Entwicklung der Informatik eine entscheidende Rolle gespielt; sie liefert Methoden und Theorien als Hilfsmittel und ist als Theoretische Informatik ein eigenständiges Gebiet der Informatik geworden. Aber mit Computern werden geistige Prozesse maschinisiert und dadurch soziale Systeme rationalisiert; die Veränderungen, die dabei geschehen, müssen wir beim Programmieren verstehen. „Unsere Hauptaufgabe ist nicht, dem Computer vorzuschreiben, was er tun soll“, sagt Donald Knuth in „Literate Programming“, „sondern anderen Menschen verständlich zu machen, was wir von ihm wollen.“ „Ein Softwaresystem lebt in seinen Entwicklern“, schreibt Peter Naur in „Programming as Theory Building“; „wenn sie weggehen, ist es tot und nicht wieder zu erwecken.“ Und wie wollen wir die Informatik verstehen, wenn wir nicht wissen, wie sie sich zu dem entwickelt hat, was sie heute zu sein scheint. Für eine Theorie der Informatik brauchen wir also neben der Mathematik andere Gebiete: Psychologie, Soziologie, Didaktik, Literatur, Linguistik, Geschichte und wie immer Philosophie. Die Problematik interdisziplinärer Arbeit, die vielen von uns zentral für eine ökologische Orientierung ist, erscheint im Sichtweisenband in der Auseinandersetzung zwischen Arno Rolf und Walter Volpert über die Frage der Gestaltung, in der Position von Peter Scheffe zur Frage der Verantwortung und in meiner

„Bederkesa-Utopie“, nach der Software nur in interdisziplinären Projekten sinnvoll entwickelt werden kann.⁵

Durch einen glücklichen Zufall war ich auf dem Weg früh auf Gregory Bateson gestoßen. Als Anthropologe hatte er gemerkt, daß er fremde Kulturen nicht aus den Eigenschaften der Menschen, sondern aus den Beziehungen der Menschen zueinander und zur Natur verstehen konnte. Tiere und Menschen entwickeln sich im täglichen Umgang, der – besonders deutlich im Spiel und im Ritual – nach festen Regeln verläuft, und etablieren dabei neue Formen. Bateson untersuchte solche Lernvorgänge, auch pathologische, und entwickelte selbst dabei eine kybernetische Lerntheorie – nachzuverfolgen in der Sammlung von Vorträgen und Aufsätzen „(Schritte zu einer) Ökologie des Geistes“. Lebewesen lernen im Wechselspiel zwischen Prozeß und Form: Sie entwickeln sich unter gegebenen Bedingungen, die sie dadurch verändern. Die Entwicklung hat eine Richtung, ist also eine Zick-Zack-Leiter. Die Paare von Form und Prozeß können beim Denken Text und Verstehen, beim Handeln Regeln und Intuition, in der Natur Leben und genetische Repräsentation sein. In seinem Alterswerk „Geist und Natur“ formuliert Bateson die Erkenntnis: Individuelle geistige Entwicklung und biologische Evolution folgen denselben Prinzipien, sie sind dasselbe auf verschiedenen Ebenen.

Will ich die These als Informatiker auf eine Theorie der Formalisierung anwenden, muß ich die Beziehung genauer formulieren. Dazu half mir wieder ein Naturerlebnis. Bei einem Urlaub in Griechenland im Herbst 1990 grübelte ich über Bateson nach; meine Frau und ich betrachteten die Fische, die in den Wellen durcheinanderspielten. Da erschienen mir meine Gedanken als Fische. Sie durchfluten mich, entstehen, vermehren sich und vergehen wieder. Beständig ist das Wissen, das sie als Schwärme, Gattungen, Arten konstituieren, und das sich nur langsam ändert. Diese lebendige Beständigkeit beruht in der Natur auf der Reproduktion über die genetische Repräsentation. Im diskursiven Denken repräsentieren wir unsere Gedanken auf dieselbe Weise durch Begriffe, die der Reproduktion und Erhaltung unseres Wissens dienen; beim Glauben und Fühlen spielen ästhetische und religiöse Werte diese Rolle.

Anschauliches und begriffliches Denken verhalten sich also zueinander wie Phänotyp und Genotyp in der biologischen Evolution. Sie sind gegensätzlich, aber gehen wechselseitig auseinander hervor. Das passt zu den didaktischen Erfahrungen mit kleinen Systemen: Wir können nicht Stoff (Begriffe, Fakten, Theorien) einschaulen und ihn dann durch Beispiele veranschaulichen; ebensowenig können wir eine Intuition erwerben und sie dann begrifflich festklopfen. Anschauung und Begriffe wachsen gemeinsam; achten wir nur auf das eine, lassen wir das andere verkümmern oder verunstalten. Beim Lehren können wir nicht einfach Wissen vermitteln, sondern lösen Lernprozesse aus; wohin sie führen, hängt vom Zufall und von den Umständen ab. Am besten lernen wir durchs Arbeiten an Aufgaben im Gespräch.⁶

Das gilt auch für das wissenschaftliche Arbeiten, sogar fürs Arbeiten mit Formalismen – wie mathematischen Theorien, logischen Kalkülen oder Programmiersprachen. Als Informatiker formalisieren wir: Wir versuchen, uns das Problem klarzumachen, legen die so entwickelte Vorstellung durch Begriffe und Regeln fest und programmieren das Ganze. Oder es liegt uns umge-

kehrt ein Programm vor, und wir versuchen zu verstehen, was wir damit bewirken können – auf dem Rechner, in einem Arbeitszusammenhang, in der Welt; wir entwickeln eine Vorstellung von dem Programm. Aber das ist in beiden Richtungen keine Einbahnstraße. Deswegen hat Christiane Floyd ein zyklisches Modell des Software-Entwurfs entwickelt, in dem die Schritte von der Anschauung zum Programm immer wieder durchlaufen werden und, zum Beispiel mit Prototyping, die endgültige Version in schrittweiser Näherung erstellt wird.⁷

Daher liegt der Bruch in der Kette, die vom lebendigen Menschen über Formalismen zur Maschine reicht, für mich nicht zwischen Anschauung und Formalem, sondern zwischen Formalismus und Maschine. Formalismen sind keine toten Regelsysteme, mit denen wir bedeutungslose Symbole manipulieren. Formale Begriffe sind durch wenige Eigenschaften festgelegt, wie gezüchtete Fische, und so wertvoll und gefährdet wie diese. Wir können mit ihnen so viel oder so wenig anfangen, wie sie uns anschaulich sind, wie wir sie intuitiv verwenden können. Tot sind sie erst im Rechner oder in der mechanischen Anwendung von Regeln per Hand, wenn wirklich nur noch Symbole geschoben werden.⁸ Auch die Aktionen des Rechners verstehen wir nur, wenn wir sie uns anschaulich machen; aber einmal programmiert, läuft er ohne unser Zutun – dann können wir uns leicht aus unserer Verantwortung stehlen.

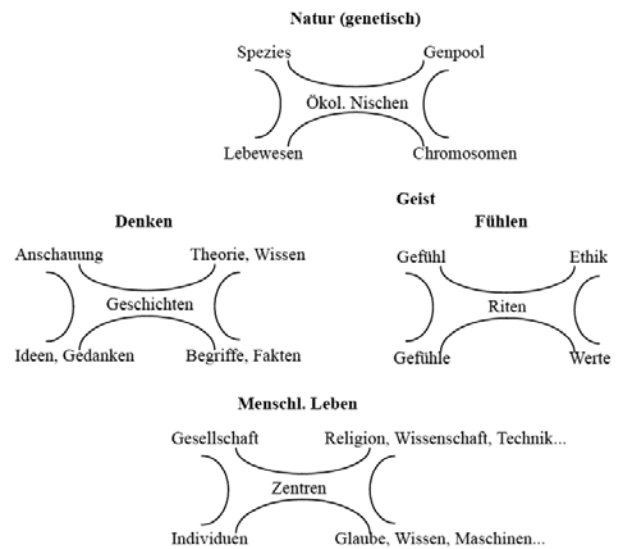
Das ist das eigentliche Problem bei der Erstellung von Software, wie es in den Beiträgen dieses Heftes (*Fiff-Kommunikation 1/1995, d. Red.*) in unterschiedlicher Form immer wieder auftaucht: Wir entwerfen eine digitale Maschine und benutzen dabei Formalismen; aber wir müssen immer die vom Rechner erstellten Ergebnisse und gesteuerten Abläufe mit dem in Einklang bringen, was wir gemäß der Situation („dem sich wandelnden Kontext“) wollen und intuitiv erwarten. Je schwieriger die Aufgabe ist, je weiträumiger das Problem oder Anwendungsfeld, desto schlimmer klafft diese Lücke. Die Lücke dadurch zu verringern, daß wir uns den zu verändernden Bereich und das zu erstellende Programm gleichzeitig vertraut machen, heißt für mich Programmieren „auf kleine Weise“.⁹ Wie immer sind solche kleinen Systeme nicht formal herstellbar, auch können sie nicht quantitativ beliebig groß werden. Eine ökologische Orientierung ist mit Softwarefabriken nicht vereinbar. Wie schon die Kleine-Systeme-Sicht ist die ökologische Orientierung eine Ethik ebenso wie eine Theorie: Wir können ein kritisches Verständnis nur im konkreten Handeln erwerben und durchhalten.¹⁰

Mit der Alternative „Fische in Schwärmen oder Fische in Dosen“ habe ich zuerst die Teilnehmer einer Konferenz des AK „Theorie der Informatik“ in Bederkesa erschreckt oder belustigt – so sehr, daß in der schriftlichen Fassung „Sinn im Formalen?“ im Sichtweisen-Band keine Fische schwimmen, sondern Spinatblätter wedeln. Im Frieden eines Forschungssemesters 1991/92 am Institute of Cognitive Studies in Berkeley und angeregt durch Seminare dort in Philosophie, Linguistik, Psychologie und Didaktik wurden daraus wieder „Fish in Schools or Fish in Cans“.¹¹ Eine Initiative des FIFF (Forum Informatiker für Frieden und gesellschaftliche Verantwortung) führte 1993 zu einem Workshop „Informationstechnik für eine lebenswerte Welt. Visionen für das 21. Jahrhundert“. Dort entstand die Idee zu einem Arbeitskreis „Ökologische Orientierung in der Informatik“, der sich in diesem Heft vorstellt. So haben die Fische eine neue Heimat ge-

funden.¹² Auf der 10. Jahrestagung des FIFF in Bremen haben Christiane Floyd, Arno Rolf und ich eine Arbeitsgruppe zu dem Thema organisiert, in der wir evolutionäre Methoden der Software-Entwicklung, Informatik-Unterstützung für eine nachhaltige ökonomische Entwicklung und ökologische Modelle in der Informatik zu einer Vorgehensweise vereinigt haben, die „die Welt heilen und heil lassen“ soll (Floyd). Meine Arbeiten zu dem Thema sind erst teilweise veröffentlicht,¹³ aber in den im Literaturverzeichnis genannten Berichten zugänglich.

Mit der Geschichte meines Weges zu einer ökologischen Orientierung in der Informatik habe ich zwei Dimensionen benannt, in deren Koordinaten ich rückblickend den Weg beschrieben habe. Die eine Dimension wird aufgespannt durch das Paar ‚Allgemeinheit – Einzelnes‘. Im Sozialen sind das Gesellschaft und Individuum; dazwischen vermitteln die kleinen Systeme, von denen die eigentlichen Wirkungen ausgehen und die so die Dimension konstituieren. Die andere Dimension wird aufgespannt durch das Paar ‚Lebendiges – Darstellung‘. So erhält und entwickelt sich die lebende Gesellschaft in den kulturellen Traditionen, wissenschaftlichen Theorien, technischen Fertigkeiten, juristischen Normen, politischen Strukturen und wirtschaftlichen Organisationen, in denen sie sich uns darstellt; so erhalten und entwickeln sich lebende Individuen in ihren Eigenschaften, Fähigkeiten und Artefakten. Was steht vermittelnd zwischen den individuellen und den gesellschaftlichen Darstellungen? Wie stellen sich kleine Systeme dar? Geistiges Leben wird in der ersten Dimension aufgespannt durch allgemeines Wissen und einzelne Gedanken, vermittelt durch die Vertrautheit in kleinen Bereichen, in denen wir uns gut auskennen. Dem entsprechen allgemeine Theorien und einzelne Fakten als geistige Form. Wieder: Was sind die kleinen Systeme dazwischen?

Als ich aus Berkeley zurückkam, empfingen mich Mitarbeiter mit einem Plan für ein Interdisziplinäres Forschungsprojekt „Sozialgeschichte der Informatik“. Wir gewannen Interessenten aus Soziologie und Hochschulforschung und -didaktik, aus Politologie und Geschichte, insbesondere Technikgeschichte, aus Psychologie und Philosophie, und die TU Berlin finanzierte das Vorhaben. Die interdisziplinäre Arbeit war schwierig, aber wurde fruchtbar. Meine bisherigen Versuche, die Informatik kritisch zu verstehen, mündeten auf erstaunliche Weise in die Fragestellungen des Projekts. Wenn ich diese Wissenschaft verstehen und verändern, wenn ich verantwortlich handeln will, muß ich fragen, wie sie sich in Verflechtung mit Technik und Kultur, mit politischen und anderen Institutionen und mit anderen Wissenschaften entwickelt hat. Im oben beschriebenen ökologischen Modell heißt das: Wissenschaftler und Wissenschaften entwickeln sich in den kleinen wissenschaftlichen Systemen, in Arbeitskreisen, Lehr- und Forschungsgruppen, intensiven Seminaren, Projekten. Und so wie beide ihre Darstellung universell und individuell in Theorien und harten Fakten, in Strukturen und deren Teilen, in Traditionen und persönlichen Fähigkeiten, in Technik und technischem Können finden, so finden wir die kleinen Systeme der Wissenschaft nur in den Geschichten, mit denen wir die jeweiligen Zusammenhänge zwischen Individuellem und Universellem und zwischen Lebendigem und seiner Darstellung beschreiben. Die Sozialgeschichte einer Wissenschaft – in der doppelten Bedeutung des Wortes Geschichte als Ereignisfolge und deren Darstellung – eignet sich in den wissenschaftlichen Gruppen und stellt sich dar in den Geschichten, die wir über sie und in ihnen erzählen. Wenn wir sehen wollen, wie eine Wissenschaft



entsteht, lebt und vergeht, müssen wir in den Fakten und Theorien, mit denen die Wissenschaftler uns abspesen wollen, die Geschichten finden und erfinden, in denen sie lebendig werden. Das ist meine Arbeitshypothese für das Sozialgeschichte-Projekt, die sich in der Projektarbeit bestätigen und modifizieren lassen muß. Sozialgeschichte wird dabei nicht zu einem Vorlesebuch wissenschaftlicher Anekdoten, so wenig wie die Welt eine Sammlung kleiner Systeme ist; aber die sozialwissenschaftlichen Theorien und Fakten entstehen aus kleinen Geschichten, so wie sich Kulturen und Individuen aus kleinen Systemen etablieren.¹⁴ Deswegen habe ich meine Vorstellungen zu einer ökologischen Orientierung in der Informatik als Geschichte meines Weges beschrieben.

Ich fasse die Wechselbeziehung der komplementären Paare ‚Einzelnes und Allgemeinheit‘ und ‚Lebendiges und Darstellung‘ in der Natur (genetische Evolution), im Geist (Denken und Fühlen) und im menschlichen Leben in vier Diagrammen zusammen, mit ökologischen Nischen, Geschichten, Riten und Zentren als jeweiliger „wirklicher Mitte“. Diagramme sind eine extreme Form formaler „Codierung“, als solche ohne den zugehörigen Text nicht verständlich, aber hilfreich zur Veranschaulichung, da sie eine klare Übersicht „auf einen Blick“ erlauben.

In den anderen Beiträgen dieses Heftes kommen scheinbar ganz andere Vorstellungen zur Sprache; die Autoren schreiben aus ihren jeweiligen Arbeitssituationen über die Aufgaben, die ihnen am dringlichsten erscheinen. Aber überall klingen die Motive an, die auch mich auf den Weg gebracht haben und die ich in der Frage zusammenfassen kann: Wie können wir als Informatiker dazu beitragen, daß die Natur lebenswert und damit die Menschen in ihr lebensfähig bleiben? Auf der Suche nach Antworten können wir an nachhaltiger Entwicklung oder menschengerechter Systemgestaltung arbeiten, die Prozesse von Arbeit und Organisation statt der Produkte in den Vordergrund stellen, Informatiker als Akteure in allen gesellschaftlichen Arenen kämpfen lassen, die Suche nach erkenntnistheoretischer und ethischer Fundierung verknüpfen, Informatik interdisziplinär ausweiten und damit zu einem neuen Verständnis von Wissenschaft kommen wollen, die Differenzen pflegen statt von Einheit auszugehen, Gestaltung als Gefahr oder als Aufgabe ansehen, optimistisch oder pessimistisch werden – wichtig ist „ökologische Orientierung“ als Metapher für eine Vision, die uns bei aller Verschiedenheit mit- statt neben- oder gar gegeneinander suchen läßt.

Mich hat die Suche dazu gebracht, die Bezeichnung „kleine Systeme“ aufzugeben. Der Austausch in den beiden Paaren des ökologischen Modells, zwischen Individuum und Gesellschaft und zwischen Leben und sozialer „Kodierung“, geschieht nicht abstrakt, sondern in kleinen sozialen Umgebungen, die ich „Zentren“ nenne. Nur wenn wir in den Zentren, in denen wir leben, diese Wechselwirkung frei gestalten, weder die Allgemeinheit noch das Einzelne, weder die Geschehnisse noch ihre Darstellungen dominieren lassen, entwickeln sie sich weiter. Solche lebendigen Zentren nenne ich „ökologisch“; mehr dazu in „Ökologische Modelle geistiger und sozialer Entwicklung“.

Referenzen

- Bammé, Arno et al 1983: Maschinen-Menschen, Mensch-Maschine. Reinbek: Rowohlt
- Bateson, Gregory 1972: Steps to an Ecology of Mind. Ballantine Books.
Deutsch: Ökologie des Geistes. Suhrkamp stw 571, 1985
- „ - 1979: Mind and Nature – a Necessary Unity. Bantam Books. Deutsch: Geist und Natur – eine notwendige Einheit. Suhrkamp 1982
- Coy, Wolfgang et al. 1992: Sichtweisen der Informatik. Wiesbaden: Vieweg
- Dillard, Annie 1974: Pilgrim at Tinker Creek. Bantam Books
- Floyd, Christiane 1994: Künstliche Intelligenz, verantwortliches Handeln. In Sybille Krämer (Hrsg.): Geist, Gehirn, Künstliche Intelligenz. DeGruyter
- „ - et al. 1992: Software Development and Reality Construction. Springer
- Knuth, Donald E. 1984: Literate Programming. The Computer Journal vol. 27, pp. 97-111
- Krämer, Sybille 1988: Symbolische Maschinen – Die Idee der Formalisierung im geschichtlichen Abriß. Wissenschaftliche Buchgesellschaft, Darmstadt
- Leopold, Aldo 1970: The Sand County Almanac. Ballantine Books Deutsche Auswahl: Am Anfang war die Erde. Wiss. Buchgesellschaft Darmstadt 1992
- Muir, John 1954: The Wilderness World of John Muir. Edwin Way Teale (ed.). Houghton Mifflin: Boston
- Naur, Peter 1985: Programming as Theory Building. In ders. „Computing – A Human Activity“, ACM Press & Addison-Wesley 1992.
- Rolf, Arno 1992: Sichtwechsel – Informatik als (gezähmte) Gestaltungswissenschaft. In Coy et al., S. 33-47
- Schefe, Peter 1992: Theorie oder Aufklärung? Zum Problem einer ethischen Fundierung informatischen Handelns. In Coy et al., S. 327-334
- Schumacher, E. F. 1974: Small is Beautiful – Economics as if People Mattered. London: Abacus. Deutsch: Rückkehr zum menschlichen Maß. Rowohlt 1977
- „ - 1979: A Guide for the Perplexed. Perennial Library. Deutsch: Rat für die Ratlosen. Rowohlt 1979
- „ - 1979: Good Work. New York: Harper Colophon Books: Deutsch: Das Ende unserer Epoche. Rowohlt 1980
- Siefkes, Dirk 1990: Formalisieren und Beweisen – Logik für Informatiker. Wiesbaden: Vieweg, 2. Aufl. 1992
- „ - 1991: Sinn im Formalen? Wie wir mit Maschinen und Formalismen umgehen. In Coy et al. 1992, S. 97-114
- „ - 1992: Fish in Schools or Fish in Cans – Evolutionary Thinking and Formalization. International Computer Science Institute Berkeley, TR-92-009, February 1992.
- „ - 1992: Formale Methoden und kleine Systeme – Lernen, leben und arbeiten in formalen Umgebungen. Wiesbaden: Vieweg
- „ - 1993: Evolutionäre Modelle in der Informatik. Technische Universität Berlin, FB Informatik, Bericht Nr. 93-15, 54 S.
- „ - 1994: Zu einer ökologischen Theorie der Informatik. Technische Universität Berlin, FB Informatik, Bericht Nr. 94-18, 40 S.
- „ - 1995: Ökologische Modelle geistiger und sozialer Entwicklung. Beginn eines Diskurses zur Sozialgeschichte der Informatik. Erscheint als Bericht des Wissenschaftszentrums Berlin für Sozialforschung

- Thoreau, Henry David 1962: Walden (and other writings). Bantam Books.
Deutsch: Walden oder Leben in den Wäldern. Detebe 19, Zürich 1997
- Varela, Francisco, Evan Thompson, and Eleanor Rosch 1991: The Embodied Mind – Cognitive Science and Human Experience. Cambridge: The MIT Press
Deutsch: Der mittlere Weg zur Erkenntnis. Scherz: Bern 1994
- Volpert, Walter 1992: Erhalten und Gestalten – von der notwendigen Zählung des Gestaltungsdranges. In Coy et al., S. 171-180
- Wagemann, Carl-Hellmut 1991: Briefe über Hochschulunterricht. Weinheim: Deutscher Studien Verlag

Anmerkungen

- 1 E.F. Schumacher: „Small is Beautiful. Economics as if People Mattered“, „A Guide for the Perplexed“, „Good Work“.
- 2 Ihre Arbeit an einer am Menschen orientierten Informatik ist am schönsten in dem Band „Software Development and Reality Construction“ zu sehen. Meine Vorträge und Aufsätze aus der Zeit sind in dem Band „Formale Methoden und Kleine Systeme“ gesammelt.
- 3 Siehe dazu mein Lehrbuch „Formalisieren und Beweisen – Logik für Informatiker“.
- 4 Siehe den Band Wolfgang Coy et al. „Sichtweisen der Informatik“, den wir gemeinsam herausgegeben haben.
- 5 Arno Rolf „Sichtwechsel – Informatik als (gezähmte) Gestaltungswissenschaft“, Walter Volpert „Erhalten und Gestalten – Von der notwendigen Zählung des Gestaltungsdrangs“, Peter Schefe „Theorie oder Aufklärung? Zum Problem einer ethischen Fundierung informatischen Handelns“, Dirk Siefkes „Sinn im Formalen? Wie wir mit Maschinen und Formalismen umgehen“. – In der Arbeit „Umdenken auf kleine Systeme“ in der Sammlung „Zu einer ökologischen Theorie der Informatik“ diskutiere ich, was die gegenwärtigen Ideen in Wirtschaft und Verwaltung zu Verschlankung, Autonomie und Gruppenarbeit mit kleinen Systemen zu tun haben.
- 6 Da habe ich viel von Carl-Hellmut Wagemann gelernt; siehe seine „Briefe über Hochschulunterricht“.
- 7 Siehe ihre Beiträge in dem schon zitierten Band „Software Development and Reality Construction“ sowie die dort oder in meinem Buch „Formale Methoden und Kleine Systeme“ zitierte Literatur.
- 8 Deswegen finde ich es unglücklich, Formalismen als „symbolische Maschinen“ zu bezeichnen, wie es Sybille Krämer in ihrem Buch dieses Titels tut. Wenn Menschen sich maschinenhaft verhalten, wie das autistische Kind in „Mensch Maschine, Maschine Mensch“ von Arno Bammé et al., bezeichnen wir sie als geistig gestört.
- 9 Siehe auch meine Arbeiten in dem Band „Formale Methoden und Kleine Systeme“.
- 10 Christiane Floyd „Künstliche Intelligenz – Verantwortungsvolles Handeln“; Dirk Siefkes „Ohne eine Theorie der Informatik keine Ethik für Informatiker, ohne eine Ethik der Informatik keine Theorie für Informatiker“ in „Evolutionäre Modelle der Informatik“.
- 11 Viel Einfluß darauf hatte „The Embodied Mind – Cognitive Science and Human Experience“ von Francisco Varela, Evan Thompson und Eleanor Rosch.
- 12 Daneben lasse ich sie in einem Nachfolgekreis des Theoriekreises die Beziehungen zwischen „Komplexität, Erfahrung und Zuverlässigkeit“ anstupfen.
- 13 Zum Beispiel in FIFF-Kommunikation 10.3, InfoTech 5.3, 5.4.
- 14 Ausführlicher in meiner Arbeit „Ökologische Theorie und kleine Systeme. Beginn eines Diskurses zur Sozialgeschichte der Informatik“.

Der Text erschien im Original in der FIFF-Kommunikation 1/1995 als Einleitung zu dem Schwerpunkt Ökologische Orientierung in der Informatik. Informationstechnik in lebendigen Zusammenhängen. Wie danken dem Autor herzlich für die Genehmigung zum erneuten Abdruck.