

Jörg Strübing

Subjektive Leistungen im Arbeitsprozeß

**Jörg Strübing**

# **Subjektive Leistungen im Arbeitsprozeß**

**Eine empirische Untersuchung von Arbeitsstilen  
in der Programmierarbeit**

**DW** Springer Fachmedien  
Wiesbaden GmbH

Die Deutsche Bibliothek — CIP-Einheitsaufnahme

**Strübing, Jörg:**

Subjektive Leistungen im Arbeitsprozeß : eine empirische  
Untersuchung von Arbeitsstilen in der Programmierarbeit /

Jörg Strübing. — Wiesbaden : Dt. Univ.-Verl., 1993

(DUV : Sozialwissenschaft)

Zugl.: Kassel, Univ., Diss., 1992

© Springer Fachmedien Wiesbaden 1993

Originally published by Deutscher Universitäts-Verlag GmbH, Wiesbaden 1993



Das Werk einschließlich aller seiner Teile ist urheberrechtlich geschützt. Jede Verwertung außerhalb der engen Grenzen des Urheberrechtsgesetzes ist ohne Zustimmung des Verlags unzulässig und strafbar. Das gilt insbesondere für Vervielfältigungen, Übersetzungen, Mikroverfilmungen und die Einspeicherung und Verarbeitung in elektronischen Systemen.

Gedruckt auf chlorarm gebleichtem und säurefreiem Papier

ISBN 978-3-663-01638-0

ISBN 978-3-663-01637-3 (eBook)

DOI 10.1007/978-3-663-01637-3

# Inhalt

<b>Vorwort</b> .....	9
<b>1. Subjektive Leistungen</b> .....	15
1.1 Subjektivität - ein erster Zugang .....	15
1.2 Widerspruchslogik.....	21
1.3 Die Logik des Arbeitsprozesses .....	27
1.4 »Habitus« als generatives Schema subjektiver Leistungen .....	35
1.5 Stil und Kultur: Erscheinungsweisen subjektiver Leistungen .....	42
1.6 Kultur als »Shopfloor culture«.....	45
1.7 Zusammenfassung und These.....	48
<b>2. Zur Struktur des Feldes Programmierarbeit</b> .....	53
2.1 Abriß zur Geschichte des Programmierens .....	55
2.2 Berufsfeld und Branchenstruktur .....	65
2.3 Softwareentwicklung als Arbeitsprozeß .....	71
2.3.1 Problemdefinition .....	72
2.3.2 Problem- bzw. Systemanalyse .....	72
2.3.3 Entwurf.....	73
2.3.4 Codierung .....	74
2.3.5 Test.....	76
2.3.6 Fehlersuche.....	77
2.3.7 Dokumentation.....	78
2.3.8 Installation.....	79
2.3.9 Wartung.....	79
2.4 Zur Arbeitsstofflichkeit in der Programmierarbeit.....	80
2.4.1 Applikation/Situierung.....	83
2.4.2 Konstitutionsproblem.....	87
2.4.3 Kontingenzproblem.....	88
2.4.4 Aushandlungsproblem.....	92
2.4.5 Planungs-/Zeitproblem.....	94
2.4.6 Zusammenfassung.....	95
2.5 Softwareentwicklung als Organisation .....	98
2.5.1 Prozeßmodelle der Softwareentwicklung.....	100
2.5.2 Arbeitsteilung und Tätigkeitsmerkmale.....	102
2.5.3 Softwareentwicklung als Projektarbeit .....	108

2.5.4 Problem der Kontrollorganisation und der Produktivität.....	110
<b>3. Arbeitsstile als Ausdruck subjektiver Aneignungsleistungen.....</b>	<b>115</b>
3.1 Die Organisation der Arbeitsschritte: Varianten einer Praxis der Durchmischung.....	118
3.1.1 Spannen der Ungewißheit verkürzen .....	119
3.1.2 Die Bedeutung von Erfahrung .....	121
3.1.3 Subjektive Präferenzen: Versinnlichung und Abwechslung .....	124
3.1.4 Störung von außen .....	127
3.1.5 Zusammenfassung.....	130
3.2 Skizzenhafte Selbstverständigung - Stile der Darstellung von Zwischenergebnissen .....	132
3.2.1 Funktionen der Darstellung von Zwischenergebnissen.....	133
Selbstveranschaulichung von Vorgaben.....	133
Sicherung von Entwurfsideen.....	135
Kollektive Bezugsbasis in Aushandlungsprozessen .....	136
3.2.2 Bezugspunkte variierender Praxen.....	138
Komplexität .....	138
Art der Arbeitsteilung/Kooperation .....	141
Abweichen von Organisationszielen .....	143
3.2.3 Zusammenfassung.....	149
3.3 Teststile: Zwischen Akribie und »auf dem Rücken fliegen« .....	152
3.3.1 Stil-Varianten.....	153
Varianten I: »Schreibitsch-« versus »Bildschirmtest«.....	153
Varianten II: Aktive und passive Begleitung der Testläufe .....	155
Varianten III: Unterschiedliche Schrittweiten beim Test .....	157
3.3.2 Arbeitsstoffliche Hintergründe .....	159
Maß erforderlicher Fehlerfreiheit .....	159
Art der technischen Hilfsmittel .....	160
Maß an Innovativität .....	162
3.3.3 Habitus und Teststil .....	164
3.4 Vom Umgang mit Tools .....	169
3.4.1 Was sind Tools?.....	170
3.4.2 Verfügbares Hilfsmittel oder normative Vorgabe .....	171
3.4.3 Arbeitslogische Unverträglichkeiten als Akzeptanzhemmnis.....	174
3.4.4 Zum Problem der generalisierenden Tendenz von Tools.....	177
3.4.5 Zusätzliche Komplexität durch Tools .....	178
3.4.6 Tools als vergegenständlichte Norm des Software Engineering .....	180
3.4.7 Ambivalenz bestimmt die Haltung der Arbeitenden .....	181
3.4.8 Unterschiedliche Gruppen: Befürworter und Kritiker .....	185
3.4.9 Zusammenfassung.....	187

<b>4. Kollektiver Habitus - Stilisierte Gruppenidentitäten und ihre arbeitsstofflichen Bezüge in der Programmierarbeit</b> .....	189
4.1 Bezugsrahmen .....	189
4.2 Schließung nach außen .....	192
4.2.1 »Softwareentwicklung« oder »Programmierarbeit« - Über die kulturelle Bedeutsamkeit der Ablehnung des Begriffs »Programmierarbeit« .....	192
4.2.2 ProgrammiererInnen und der »Habitus der Geldwelt«: Konträre Habitus und ihre Entstehungsgründe .....	194
4.2.3 »Ein Strich unter dem Namen« - Die Abgrenzung von der Hierarchie der Vorgesetzten und eigene Aufstiegsambitionen .....	201
4.2.4 »Kleider machen Leute«: Die Kleider(un)ordnung als Stilmittel der Abgrenzung .....	210
4.2.5 Was Programmierer nicht sind: Schreiber .....	215
4.3 Differenzierung nach innen .....	220
4.3.1 Ingenieure, Informatiker und andere: Verschiedene Berufsgruppen in einem Team .....	221
4.3.2 Von »Hackern« und »Tüftlern« in der Programmierarbeit .....	225
4.3.3 Männer und Frauen: Harte und weiche Programmierstile? .....	233
4.4 Integrationsagentur Projektteam: Trainig-on-the-job als Initiationsritus .....	236
4.5 Zusammenfassung .....	241
<b>5. Subjektive Leistungen und ihre Bedeutung in der Programmierarbeit</b> .....	243
5.1 Zu einer typologischen Skizze von Programmierarbeit und ProgrammiererInnen .....	244
5.1.1 Arbeitsstile: Die konkrete Gestalt subjektiver Leistungen .....	244
Zu a: Die Organisation der Arbeitsschritte .....	246
Zu b: Die Darstellung von Zwischenergebnissen .....	248
Zu c: Test und Fehlersuche .....	250
Zu d: Nutzung von Softwaretools .....	254
Versuch einer Generalisierung .....	258
5.1.2 Die ProgrammiererInnen - Skizze eines kollektiven Habitus .....	266
5.1.3 Alternative Entwürfe, variierende Arbeitsweisen in der Softwareentwicklung konzeptuell einzubinden .....	273
Boundary Spanning - ein mechanistischer Kommunikationsbegriff .....	273
»Opportunistic thoughts« zur Bewältigung von »ill structured problems« .....	276

»Stille Leistungen« - eine Variante informellen Arbeitshandelns .....	279
<b>5.2 Arbeitspolitische Aufschlüsse aus der Analyse subjektiver Leistungen .....</b>	<b>281</b>
5.2.1 Subjektive Leistungen als Quelle organisationeller Lernprozesse .....	281
5.2.2 Rationalisierung: Potentiale und Schranken .....	286
Grenzen der Arbeitsteilung .....	286
Flexibilität als Gestaltungserfordernis für Arbeitsmittel .....	288
Produktivitätspotential »Mensch« .....	289
5.2.4 »Politisierung der Betriebsstruktur«? .....	292
<b>5.3 Schlußbetrachtungen .....</b>	<b>295</b>
5.3.1 Erfahrungen mit den verwendeten theoretischen Konzepten .....	296
5.3.2 Ein methodisches Resümee .....	299
5.3.3 Aufgaben zukünftiger Forschungen zu subjektiven Leistungen in der Programmierarbeit .....	302
<b>Anhang: Zur Durchführung der Untersuchung .....</b>	<b>305</b>
A.1 Ausgangsüberlegungen .....	305
A.2 Methodendiskussion .....	306
A.3 Gegenstand der Befragung .....	309
A.4 Die Auswahl der Befragten .....	312
A.5 Das Sample im einzelnen .....	314
A.6 Die Auswertung .....	318
A.7 Interviewleitfaden für ProgrammiererInnenbefragung 1989/90 .....	321
<b>Literatur .....</b>	<b>325</b>

# Vorwort

Am Beginn dieser Untersuchung stand eine andere: Als ich 1986/87 meine Forschungen zur Programmierarbeit begann, ging es mir recht allgemein darum, einen »weißen Fleck« auf der arbeitssoziologischen Landkarte mit ersten Konturen zu füllen. Die dazu damals durchgeführten Intensivinterviews haben jedoch nicht allein diesen Zweck erfüllt, die detaillierte Darstellung der Praxis von ProgrammiererInnen förderte zugleich eine Reihe von Aspekten zutage, die eine genauere Untersuchung lohnenswert erscheinen ließen: Auffällig oft betonten die Befragten, daß die Art, wie sie dies oder jenes bearbeiteten, ihren persönlichen »Stil« darstelle, daß ihnen einige Aspekte ihrer Arbeitstätigkeiten mehr oder weniger »lägen« oder daß man in ihrer Gruppe, verglichen mit anderen, eine spezielle Art habe, bestimmte Aufgaben zu bewältigen. Im Kontext meiner damaligen, eher explorativen Fragestellung konnte ich diesen Hinweisen nicht mit der gebotenen Sorgfalt nachgehen, zumal derartige Äußerungen eher sporadisch im Material auftauchten und dieses hinsichtlich einer so speziellen Untersuchungsfrage nicht reichhaltig genug schien, um mehr als vage Hypothesen zu äußern.

Eine dieser Hypothesen lautete, daß solch individuelle oder kollektive Eigenheiten in den Arbeitsweisen weder zufälliger Natur, noch für die Qualität der Aufgabenbewältigung neutral sind, daß sie vielmehr mit der Struktur der jeweiligen Arbeitssituation im Zusammenhang stehen und ihnen ein spezifischer *Leistungscharakter* eignet. Eine zweite Untersuchung mit erweiterter empirischer Basis sollte nun genauer klären, wie der Strukturzusammenhang beschaffen ist, auf den variierende Arbeitsweisen sich beziehen und welcher Art die Leistungen sind, die damit erbracht werden. Die Ergebnisse dieser Untersuchung sind Gegenstand der vorliegenden Arbeit.

Dabei wird sich zeigen, daß die »Arbeitsstile« der untersuchten SoftwareentwicklerInnen primär *Vermittlungsleistungen* darstellen: In ihnen beziehen die Akteure heterogene Strukturdimensionen der Organisation wie des Arbeitsgegenstandes aufeinander, interpretieren sie situationsbezogen, und stellen damit nicht nur ein Arbeitsprodukt her, sondern zugleich auch erst die Arbeitsaufgabe selbst in ihrer vollständigen Form sowie den fakti-



schen Arbeitsprozeß. Indem sie dies tun, setzen sie aber zugleich sich selbst als Personen ins Verhältnis zu den Arbeitsanforderungen, sie eignen sich diese an und werden zugleich von ihnen angeeignet. Gerade weil Arbeitsanforderungen unvollständig und zu Teilen gar in sich widersprüchlich sind, bedürfen sie der deutenden, ergänzenden, entscheidenden und vermittelnden *Eigentätigkeit* der Subjekte.

Die Arbeitenden können also gar nicht »als Marionetten an den Fadenenden der Formalorganisation« (Ekardt u.a. 1988: 34) gehen, sie halten vielmehr ein Ende jenes Fadens in der Hand, aus dem der Arbeitsprozeß gesponnen wird. Ließen sie los, käme ein solcher nicht mehr zustande. Diese Leistungen der Akteure, die sich in ihren Arbeitsstilen, aber auch in den darüber hinausgehenden, vielfältigen arbeitskulturellen Formen im »Shopfloor« der Programmierarbeitsteams manifestieren, nenne ich »subjektive Leistungen«. Auf sie sind Arbeitsorganisationen um ihres Bestands willen angewiesen, wie andererseits die Arbeitenden ihrer bedürfen, um Identität zu entfalten. Dies heißt nun wiederum nicht, daß Subjektivität im Arbeitsprozeß sich in purer Funktionalität erschöpft. Der Sinn, den die EntwickleInnen ihrem Handeln geben, steht immer wieder auch im Konflikt mit übergreifenden Organisationszielen, wie auch mit Partialinteressen anderer Beteiligter, er ist perspektivisch gebunden, gewinnt seine Gestalt vor dem Hintergrund jeweiliger »subjektiver Dispositionen« (Bourdieu) und weist damit zugleich über die Arbeit hinaus.

In fünf Schritten stelle ich die Ergebnisse meiner Untersuchung dar. Das *erste Kapitel* dient dabei zunächst einer theoretischen Annäherung an einen tragfähigen Begriff von Subjektivität und »subjektiven Leistungen« (1.1). Es werden verschiedene Begriffsfassungen auf ihre Brauchbarkeit für die Analyse von Arbeitsprozessen befragt, und es wird schließlich vorgeschlagen, Subjektivität als Vermittlungsbegriff im vorgenannten Sinne zu fassen. Sodann führt dieses Kapitel in drei soziologische Konzepte ein, die sich im Gang meiner Untersuchung als besonders erklärungsfähig erwiesen haben (1.2 bis 1.4), bevor schließlich »Arbeitsstil« (1.5) und »Shopfloor culture« (1.6) als empirische Ausdrucksformen »subjektiver Leistungen« genauer gefaßt werden. Das Ende dieses ersten Darstellungsschrittes bildet die Formulierung einer These zum Stellenwert »subjektiver Leistungen« im Arbeitsprozeß des Programmierens (1.7).

Während ich mich im ersten Kapitel mit soziologischen Grundfragen beschäftige, wende ich mich im *zweiten Kapitel* dem Feld zu, das meine Untersuchung zum Gegenstand hat. Programmierarbeit ist ein noch »junger«

---

Zweig gesellschaftlicher Arbeit, dem allerdings in mehrfacher Hinsicht besondere Bedeutung für deren Zukunft zukommt. Wenngleich mit Raymundus Lullus ein erster Vordenker der Computertheorie schon im späten Mittelalter zu finden ist (Künzel/Cornelius 1987) und Charles Babbage mit seiner »analytischen Maschine« bereits vor fast anderthalb Jahrhunderten den Prototyp einer universell programmierbaren Maschine vorstellte (Hyman 1987), ist doch die Technikgeschichte des elektronischen Rechners kaum 50 Jahre alt und die Sozialgeschichte der erwerbsmäßigen Entwicklung speicherresidenter Programme gerade gut 40 Jahre. Zugleich aber ist unübersehbar, daß sowohl die Produkte dieses Erwerbszweiges innerhalb kürzester Zeit in fast alle Bereiche des gesellschaftlichen Lebens eingedrungen sind, als auch daß viele Bereiche der Erwerbsarbeit heute oder in naher Zukunft wenigstens partiell mit Tätigkeitsformen befaßt sein werden, wie sie bislang für Programmierarbeit kennzeichnend waren. Deren junge Geschichte wird im zweiten Kapitel ebenso skizziert (2.1) wie die aktuellen Strukturen der Datenverarbeitungsbranche und des Berufsfeldes der SoftwareentwicklerInnen (2.2). Erst auf dieser Grundlage wende ich mich dem Arbeitsprozeß des Programmierens selbst zu, und dies in drei Etappen: Zunächst werden die verschiedenen zur Programmierung erforderlichen Arbeitsschritte kurz und aus der Sicht der Informatik erläutert (2.3), bevor ich in eine intensivere Diskussion der »Arbeitsstofflichkeit« (Ekardt u.a.) von Programmierarbeit eintrete (2.4) und schließlich einen Überblick über die Probleme ihrer Organisation gebe (2.5). Gerade dieser letzte Abschnitt sollte deutlich machen, welche Bedeutung der sozialwissenschaftlichen Auseinandersetzung mit Programmierarbeit auch für die Branche selbst zukommt: Der hohe Anteil »subjektiver Leistungen« in der Softwareentwicklung ist dort zwar nicht unbemerkt geblieben, wird aber bevorzugt als Mißstand beklagt, statt als Produktivkraft erkannt und gefördert zu werden. Dabei ist das Lamento des Softwaremanagements, das »what's wrong with computer programming management« (Reynolds 1970), ungefähr so alt wie die Branche, und alle Bemühungen des Software Engineering konnten der »Softwarekrise« bislang nicht entscheidend beikommen.

Die beiden ersten Kapitel bilden - ein jedes auf seine Weise - die Grundlage zum Verständnis der beiden folgenden, in denen ich Auszüge aus dem empirischen Material in ihrem Kontext darstelle und analysiere. Das *dritte Kapitel* befaßt sich dabei mit den Arbeitsstilen der Befragten. Hier wird allerdings nicht der gesamte Prozeß der Programmentwicklung untersucht, sondern vier exemplarische Ausschnitte: die Organisation der Arbeitsschrit-

te (3.1), die Darstellung von Zwischenergebnissen (3.2), die Arbeit des Testens und der Fehlersuche (3.3) sowie der Umgang mit softwaretechnischen Arbeitsmitteln (3.4). Analysiert werden jeweils die von den Befragten praktizierten Arbeitsweisen im Kontext fallspezifischer arbeitsstofflicher Hintergründe und organisationsseitiger Vorgaben. Hier wird deutlich werden, im Rahmen welcher Handlungsalternativen die Akteure ihre Praxis gestalten und auf welche Relevanzstrukturen sie sich dabei implizit beziehen oder explizit berufen. Dabei lassen sich zugleich verschiedene Aspekte konkret erbrachter »subjektiver Leistungen« ausmachen.

Betont das dritte Kapitel aus analytischen Gründen stärker die Perspektive individueller Aufgabenbewältigung, so wendet sich das *vierte* gezielt kollektiven Aspekten der Programmierarbeit zu. Untersucht werden stilisierte Gruppenidentitäten der Berufsgruppe der SoftwareentwicklerInnen insgesamt (4.2), aber auch deren interne Differenzierungen (4.3). Hier beschreibe ich kollektive »Habitus« (Bourdieu), in denen sich EntwicklerInnen von anderen am Programmentwicklungsprozeß beteiligten Gruppen abgrenzen und gehe vor allem der Frage nach den arbeitsprozessualen Ursprüngen dieser Abgrenzungen nach - womit wir auf einem Umweg wiederum zu Aufschlüssen über »subjektive Leistungen« gelangen, die einen Teil des »Materials« darstellen, aus dem derartige Gruppenidentitäten sich speisen. Der letzte Abschnitt dieses Kapitels schließlich geht der Frage nach der Genese kollektiver Habitus nach (4.4).

Im abschließenden *fünften Kapitel* fasse ich zunächst die in den beiden vorhergehenden Kapiteln ausgebreiteten Ergebnisse meiner Analysen zu einer typologischen Skizze »subjektiver Leistungen« und ihrer Bezugsdimensionen in der Programmierarbeit zusammen und diskutiere einige Charakteristika des kollektiven Habitus von SoftwareentwicklerInnen (5.1). Im Anschluß an einen kurzen Vergleich meines Konzeptes mit anderen Ansätzen, subjektive Aspekte in der Programmierarbeit systematisch zu fassen, wende ich mich einer ausführlichen Diskussion der arbeitspolitischen Aufschlüsse zu, die sich aus den Ergebnissen meiner Studie ziehen lassen (5.2). Dabei diskutiere ich die beschriebene und analysierte Praxis der EntwicklerInnen vor allem als mögliche Quelle organisationeller Lernprozesse: In den Arbeitsstilen der EntwicklerInnen wird nicht nur sichtbar, daß der Kontrollier- und Planbarkeit dieser Arbeit ebenso objektive Grenzen gesetzt sind, wie ihrer Dekomposition und Technisierung. Es zeigt sich auch, daß die Arbeitenden selbst die zentrale Produktivkraft in der Programmierarbeit sind. Eine Produktivkraft indes, mit der umzugehen der Wissenschaft Informatik wie

---

auch der Managementpraxis offenbar nicht ganz leicht fällt. In einem Resümee am Ende des fünften Kapitels behandle ich drei Aspekte: Zunächst diskutiere ich die praktischen Erfahrungen, die ich im Laufe der Untersuchung bei der Verwendung der drei im ersten Kapitel eingeführten soziologischen Konzepte gewinnen konnte. Im Anschluß daran gilt es, ein forschungsmethodisches Fazit zu ziehen und dabei auf die Grenzen hinzuweisen, die mein methodischer Zugang der Verallgemeinerbarkeit der Ergebnisse setzt. Abschließend stelle ich dar, welche Fragen meine Arbeit unbeantwortet lassen muß und zeichne daraus ein Panorama zukünftiger Forschungsaufgaben.

Wissenschaft pflegt - zumindest sollte sie dies - einen methodisch kontrollierten Wirklichkeitszugang, und so hat jede Untersuchung über die Einhaltung dieser Regeln Rechenschaft zu geben. Die diesbezügliche Erörterung einschließlich einer Beschreibung des der Studie zugrundeliegenden Samples findet sich im Anhang.

Forschung kostet Geld. Eine Binsenweisheit, die dennoch Anlaß zu dem Hinweis gibt, daß diese Untersuchung sich der Förderung aus zentralen Forschungsmitteln der Gesamthochschule Kassel sowie aus dem Landeschwerpunktprogramm Technikfolgenforschung des Landes Hessen im Rahmen der Forschungsgruppe »Rationalität des Ingenieurhandelns« erfreute, und ohne diese kaum zustande gekommen wäre. Doch Geld allein macht nicht nur nicht glücklich, es trägt auch nur begrenzt zum Erkenntnisgewinn bei. In dieser Hinsicht habe ich vor allem Werner van Treeck, Friedrich Weltz und Hans-Peter Ekardt zu danken, die mir mit ihren Erfahrungen in diesem Feld soziologischer Forschung jederzeit anregend zur Seite standen. Zu danken habe ich aber auch Christoph Oehler, der mich beständig zur Weiterarbeit an meinem Thema ermutigte und Christoph Gillessen, der sich als unerschrockener Erstleser aller Rohfassungen erhebliche Verdienste um das Wohl der geneigten Leserschaft erworben hat. Hervorgehoben werden muß weiterhin die Hilfe, die mir Sabine Stange in den verschiedenen Etappen des Forschungsprozesses hat angedeihen lassen. Sie hat nicht allein die transkribierten Interviewtexte der EDV zugänglich gemacht und so manche Stunde bei Literaturrecherchen in der Bibliothek zugebracht, sondern ist auch mit gespitztem Rotstift dem Heer der Tipp- und Rechtschreibfehler zu Leibe gerückt. Weitere Literaturrecherchen sowie eine Vorauswertung des Materials erstellte Jörg Daniel, die oft mühsamen Transkriptionen besorgten Angelika Flörke, Helga Cassidy und Sybille Liebrecht. Ihnen gilt mein Dank ebenso, wie all jenen Freundinnen und Freunden, die in den letzten

Jahren meine Launen ertragen mußten, und die ich oft genug vernachlässigt habe. Das Ende dieser schier unendlichen Danksagung ist ein besonderer Platz. Er bleibt meinen Gesprächspartnerinnen und Gesprächspartnern im Feld vorbehalten, die mich in stundenlangen, meist in der Freizeit geführten Interviews Einblick in ein faszinierendes Feld nehmen ließen.