Andrea Baukrowitz · Andreas Boes · Bernd Eckhardt

Software als Arbeit gestalten

Konzeptionelle Neuorientierung der Aus- und Weiterbildung von Computerspezialisten

Andrea Baukrowitz
Andreas Boes
Bernd Eckhardt

unter Mitarbeit von
Christian Boß, Ulrich Hütten, Ulrich Jung,
Kai Michelsen und Margund Zetzmann

Software als Arbeit gestalten

Konzeptionelle Neuorientierung der Aus- und Weiterbildung von Computerspezialisten

Vorwort

Die Informationstechnik- (IT-)Fachkräfte sind zu einer Schlüsselgruppe für den wirtschaftlichen und sozialen Fortschritt geworden. Mit dem Vordringen der Computer in den letzten drei Jahrzehnten hat diese sehr junge Berufsgruppe einen enormen Bedeutungszuwachs erfahren. Der Beitrag der Computerspezialisten für die Wettbewerbsfähigkeit der Unternehmen und ebenso für die sozialverträgliche Gestaltung der Arbeit wächst in dem Maße, wie sich die Computertechnologie nicht mehr aus unserem Leben wegdenken läßt.

Die beruflichen Anforderungen, die an die Computerfachleute gestellt werden, wechseln rasch. Alle zwei bis fünf Jahre ändert sich die technologische Basis ihrer Arbeit. Lebenslanges Lernen ist unverzichtbar. Die Weiterbildungsaktivitäten in dieser Berufsgruppe sind deshalb überdurchschnittlich hoch.

Der Anforderungswandel heute ist Folge der aktuellen Umbruchprozesse in den Unternehmen. Moderne Innovationsprozesse weisen den IT-Fachkräften eine neue Rolle zu. Um diesen Anforderungen gerecht zu werden, müssen die technischen Qualifikationen der Computerspezialisten mit sozialen Qualifikationen verbunden werden. Ein neues Qualifikationsprofil ist gefordert.

Die Gestaltung moderner informationstechnischer Systeme erfordert heute von IT-Fachkräften die Fähigkeit, technisches Know-how in die Innovationsprozesse zur Entwicklung neuartiger Unternehmens- und Arbeitsstrukturen einzubringen. Ganzheitlich Qualifizieren heißt, die Qualifikation zur Gestaltung technischer Systeme mit der Kompetenz zur Kooperation mit anderen Experten und der Fähigkeit zur Bewältigung komplexer sozialer Prozesse zu verschmelzen.

Im Forschungs- und Entwicklungsprojekt »Qualifizierungsziel Ganzheitliche Arbeitsgestaltungskompetenz. Sozialverträgliche Gestaltung von IuK-Systemen als Gegenstand der Aus- und Weiterbildung von IT-Fachkräften« wurde ein modernes Qualifizierungskonzept für IT-Fachkräfte entwickelt und erprobt, das diesen Anforderungen entspricht. Im Mittelpunkt des Projektes stand die Neuorientierung der Qualifizierung auf ein ganzheitliches Qualifizierungsziel. An die Stelle des »Technikers« tritt die Ganzheitliche Arbeitsgestaltungskompetenz als Leitbild für die konzeptionelle Ausrichtung der Aus- und Weiterbildung. Qualifizierung soll zukünftig ein modernes, integriertes Verständnis der Aufgaben von IT-Fachkräften vermitteln.

Besonderer Wert wurde seitens des Auftraggebers darauf gelegt, die Projektergebnisse von Beginn an eng an den praktischen Erfordernissen in der Aus- und Weiterbildung zu orientieren. Die Grundlage zur Realisierung dieses Anspruchs

war die gute Zusammenarbeit mit unseren Kooperationspartnern: dem Bildungszentrum für informationsverarbeitende Berufe (b.i.b.), der Gesellschaft für Mathematik und Datenverarbeitung (GMD) und der Henkel KGaA. Unser besonderer Dank gilt den Mitarbeitern dieser Institutionen, insbesondere Herrn Leuthäusser, Herrn Meyer, Herrn Schröder und Herrn Willeke für das b.i.b., Frau Esser, Frau Dr. Lukat, Frau Theidig, Herrn Börding, Herrn Hüttenhain und Herrn Seidel für die GMD sowie Herrn Dr. Heymann, Herrn Minx und Herrn Welscheidt für die Henkel KGaA, die durch ihr Engagement für die notwendige »Bodenhaftung« des Projektes gesorgt haben.

Dieses Projekt wäre ohne die Förderung durch das Ministerium für Arbeit, Gesundheit und Soziales des Landes Nordrhein-Westfalen nicht möglich gewesen. Denn obwohl die Qualifizierung generell und die von IT-Fachkräften im besonderen als durchaus wichtige Frage gilt - und übereinstimmend großer Innovationsbedarf gesehen wird -, tut man sich allenthalben schwer damit, die notwendigen Entwicklungskosten aufzubringen. Dies gilt umso mehr, als der Wandel der beruflichen Anforderungen an IT-Fachkräfte eine grundlegende Neuorientierung der Qualifizierung notwendig macht. Denn immer da, wo bekanntes Terrain verlassen werden muß, wird es schwer, die Finanzierung notwendiger Entwicklungsaktivitäten zu rechtfertigen. Hier die notwendigen Entwicklungskosten zu tragen, damit die schwierigen ersten Schritte hin zu ausstrahlungskräftigen Modellen gegangen werden können, gehört zu den Aspekten der Philosophie des Landesprogramms »Mensch und Technik - Sozialverträgliche Technikgestaltung«, deren Berechtigung sich am Beispiel dieses Projektes gut nachvollziehen lassen. Dabei sind wir uns bewußt, daß dieses Projekt sein letztendliches Zustandekommen und das praxisorientierte Design den vielen Anregungen zu verdanken hat, die wir von den Verantwortlichen für das Programm, Herrn Riepert auf Seiten des Ministeriums für Arbeit, Gesundheit und Soziales sowie Herrn Loss auf Seiten des Projektträgers im Institut für Arbeit und Technik erhalten haben. Ihnen und Frau Schönberg-Sakouhi gilt unser besonderer Dank für die Unterstützung und das große Engagement, das sie diesem Projekt entgegengebracht haben.

Andrea Baukrowitz Andreas Boes Bernd Eckhardt

Inhaltsübersicht

Einleitung 19

Teil A
Qualifizierungsziel »Ganzheitliche Arbeitsgestaltungskompetenz«

I Zur Entwicklung des Qualifikationsprofils von IT-Fachkräften 25

II Didaktische Fundierung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« 52

III Didaktisch-methodische Umsetzung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« 143

Teil B
Rahmencurriculum »Moderne Software-Entwicklung«

Einleitung 243

I Mensch - Arbeit - Computer 248

II Das Software-Projekt 265

III Software-Entwicklung und Arbeitsgestaltung 286

IV Software-Entwicklung als kooperative Gestaltungsaufgabe 326

V Prozeßsteuerung in der Software-Entwicklung 379

Teil C
Qualifizierungsbaustein »Aufgabenverständnis«.
IT-Fachkräfte im Spannungsfeld zwischen
Technikgestaltung und Arbeitsgestaltung

Einleitung 409

AE 1 Einstiegsbaustein »Software Engineering« 413

AE 2 Die Entwicklung der Informationstechnik 416

AE 3 Aufgaben von IT-Fachkräften 422

AE 4 Aufgabenverständnis und Kompetenzen von IT-Fachkräften 439

AE 5 Ausgang »Methoden des Software Engineering« 452

Literatur 455

Inhalt

Einleitung 19

Teil A

Qualifizierungsziel »Ganzheitliche Arbeitsgestaltungskompetenz«

I Zur Entwicklung des Qualifikationsprofils von IT-Fachkräften 25

1 Was sind IT-Fachkräfte? 25

2 IT-Fachkräfte an der Schwelle zu einer neuen Qualifikation 33

2.1 IT-Gestaltung und veränderter Nutzungskontext 34

2.2 Arbeitsgestaltung als Bezugssystem der IT-Entwicklung 39

3 Ganzheitliche Arbeitsgestaltungskompetenz als Qualifikationsprofil der Zukunft 46

II Didaktische Fundierung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« 52

Vorbemerkung 52

1 Der besondere Charakter des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« und daraus folgende Schwierigkeiten der Vermittlung 55

2 Ansatzpunkte einer Strategie der Vermittlung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« 56

2.1 Die notwendige Unterscheidung zwischen Arbeitsprozeß und Arbeitstätigkeit 56

2.2 Der Unterschied und die Berührungspunkte des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« und des Konzepts der Schlüsselqualifikationen 58

2.2.1 Das Dilemma des Konzepts der Schlüsselqualifikationen - eine notwendige Abgrenzung 58

2.2.3 Die Wiederbelebung der didaktisch-methodischen Diskussion durch das Konzept der Schlüsselqualifikationen 60

2.3 Bestimmung der Grundelemente einer Didaktik der Ganzheitlichen Arbeitsgestaltungskompetenz - die Reformulierung von »Fachlichkeit« und »Beruflichkeit« 64

2.3.1 Die Scharnierfunktion der didaktischen Grundelemente »Beruflichkeit« und »Fachlichkeit« zwischen Arbeitsprozeß und Unterricht 64

2.3.2 Probleme der Neubestimmung der »Fachlichkeit« und »Beruflichkeit« von IT-Fachkräften 67

2.4 Das »Dreieck beruflicher Handlungskompetenz« als didaktischer Ansatzpunkt zur Umsetzung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« 78

2.4.1 Berufliche Handlungskompetenz als analytischer Begriff 82

2.4.2 Drei Dimensionen der beruflichen Handlungskompetenz und die Erfordernisse der Ganzheitlichen Arbeitsgestaltungskompetenz 86

3 Wissen als Prozeß 88

3.1 Wissen als Prozeß und seine Bedeutung im Konzept der Ganzheitlichen Arbeitsgestaltungskompetenz 91

3.2 Vom algorithmisierten Wissen zum heuristischen Wissen 94

3.3 Vom schematischen zum multiperspektivischen Wissensumgang 96

4 Personengebundene soziale Kompetenzen 97

4.1 Die Funktionalität sozialer Kompetenzen 98

4.2 Eine Systematik sozialer Kompetenzen 99

4.2.1 Frustrationstoleranz 100

4.2.2 Ambiguitätstoleranz 100

4.2.3 Rollendistanz 101

4.2.4 Empathie 103

4.2.5 Sprachkompetenz 104

4.3 Soziale Kompetenzen und die Handlungsbereiche der Ganzheitlichen Arbeitsgestaltung 108

5 Berufliche Identität und Handlungskompetenz 113

5.1 Drei Aspekte der Identität 114

5.2 Rollenidentität, personale Identität, Ich-Identität und berufliche Identität 116

5.3 Die berufliche Identität von IT-Fachkräften 122
5.3.1 Der »Hacker« 123
5.3.2 Der »rationale Techniker« 125
5.4 Ganzheitliche Arbeitsgestaltungskompetenz und die notwendige Transformation beruflicher Identität 128
5.4.1 Der »Prozeßmoderator« - Von der technischen Verantwortung zur Prozeßverantwortlichkeit 130
5.5 Identitätsbildung als Integration von Vorerfahrungen 134
6 Zwischenbetrachtung: Die Analyse beruflicher Handlungskompetenz und didaktisch-methodisches Handeln 140

III Didaktisch-methodische Umsetzung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« 143

1 Einleitung 143
1.1 Ein Paradigmenwechsel in der Qualifizierung 143
1.2 Das Problem der Durchsetzung des Paradigmenwechsels 144
1.3 Kommunikationskultur und Organisationskultur als Voraussetzungen zur Umsetzung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« 145
1.4 Unser Beitrag zur Selbstorganisation des Paradigmenwechsels 147
2 Von den Schwierigkeiten, Dozent zu sein 148
2.1 Abschied von der didaktischen Maschine 150
2.2 Der Lehrer als »exemplarischer Intellektueller« 153
2.3 Das Paradigma der »Beschleunigungsdidaktik« und ihr »heimlicher Lehrplan« 157
2.4 Heuristische Modelle als Hilfen der Handlungsorientierung in einer durch Unsicherheiten geprägten Situation 159
2.4.1 Lernaufgaben und Lernfähigkeiten 162
2.4.2 Lernhilfen und Lernhemmungen 163
3 Ganzheitliche Arbeitsgestaltungskompetenz und die didaktische Gestaltung des Unterrichts 165
4 Didaktisches Handeln und Wissen 167
4.1 Problemorientiertes Gestalten der Lernaufgabe 169
4.2 Wissensvermittlung und Wissensmoderation 174
4.2.1 Wissensvermittlung: Zwischen Aneignung und Beschleunigungsdidaktik 175

4.2.2 Wissensvermittlung: Zwischen Medialisierung und direkter Kommunikation 177
4.2.3 Wissensvermittlung: Zwischen den Formen des Wissens 180
4.2.4 Wissensmoderation: Zwischen nomologischer Perspektive und Kreativität 182
4.2.5 Wissensmoderation: Zwischen Analogien und Metaphern - Grenzen und Möglichkeiten der Wissensmoderation von Alltagserfahrungen 185
4.2.6 Wissensmoderation: »Moderation aus der Sackgasse« oder der Irrtum zwischen Zufall und System 186
4.3 Zusammenfassung: Ganzheitliche Arbeitsgestaltung und der didaktische Umgang mit Wissen 188
5 Didaktisches Handeln und das Training sozialer Kompetenzen 189
5.1 Kommunikationsregeln in der Gruppenarbeit 190
5.1.1 Kommunikationsregeln als didaktische Methode der Negation 192
5.1.2 Pragmatische Axiome der Kommunikation und die vier Aspekte einer Nachricht 194
5.1.3 Die pragmatischen Axiome der Kommunikation und die didaktische Förderung sozialer Kompetenzen durch Reflexion und Kommunikationsregeln 197
5.2 Unterrichtsformen und soziales Lernen 207
5.2.1 Kooperative Unterrichtsformen zur systematischen Förderung sozialer Kompetenzen - Unterrichtsform und Arbeitsprozeß 207
5.2.2 Die Antizipation »evidenter sozialer Lernsituationen« als Planungselement bei kooperativen Unterrichtsformen 210
5.3 Zusammenfassung: Didaktisches Handeln und soziale Kompetenzen 215
6 Didaktisches Handeln und die Entwicklung der beruflichen Identität 217
6.1 Negative Formen der Identitätsentwicklung als Nebenfolgen der Lernorganisation und der Lernformen - zur Hegemonie des Programmcodes in der Ausbildung 218
6.2 Lernorganisation und Lernformen als Ausgangspunkt einer adäquaten beruflichen Identität 223
6.3 Didaktisches Handeln, das direkt zur Entwicklung einer unangemessenen Identität beiträgt 229
6.4 Didaktisches Handeln, das die Entwicklung einer angemessenen beruflichen Identität unterstützt - fundamentale Ideen im didaktischen Kontext 233

6.5 Zusammenfassung: Didaktisches Handeln und die Entwicklung der beruflichen Identität....238

7 Resümee: Subjektorientierte Didaktik und Ganzheitliche Arbeitsgestaltungskompetenz 239

Teil B

Rahmencurriculum »Moderne Software-Entwicklung«

Einleitung 243

I Mensch - Arbeit - Computer 248

1 Probleme der Software-Entwicklung 248

2 Software im Arbeitsprozeß 252

2.1 Die allgemeinen Bestimmungsmomente der Arbeit 253

2.1.1 Arbeitsgegenstand Information 257

2.1.2 Arbeitsmittel Software 260

2.2 Neue Unternehmensstrategien und Software-Einsatz 262

3 Anforderungen an die Software-Entwicklung 263

II Das Software-Projekt 265

1 Einführung 265

2 Software und Software-Qualität....266

3 Arbeitsgestaltung als Ziel des Software-Projekts 270

4 "To program is to understand!" - Kooperation als Notwendigkeit 273

5 Lernprozesse und Projektorganisation 278

6 Kooperation im Spannungsfeld betrieblicher Interessen 280

7 Lernen und Konsens im Prozeß 282

8 Zusammenfassung 284

III Software-Entwicklung und Arbeitsgestaltung 286

1 Arbeitsgestaltung im Software-Projekt 287
1.1 Leitbilder der Software-Entwicklung 287
1.2 Ziele der Arbeitsgestaltung im Software-Projekt 290
2 Die Analyse des Arbeitsprozesses 297
2.1 Unternehmensstrukturen und Arbeitsprozesse 297
2.2 Die Rahmenbedingungen des Arbeitsprozesses 299
2.3 Die Ist- Analyse 301
3 Die Gestaltung des Arbeitsprozesses 302
3.1 Das Leitbild der »vollständigen Tätigkeitsstruktur« 303
3.2 Die arbeitsplatzbezogene Gestaltung von Arbeit 306
3.2.1 Die Gestaltung vollständiger Aufgabenbereiche 307
3.2.2 Die Handlungsregulation als Ansatzpunkt der Software-Unterstützung 308
3.3 Gestaltung der Kooperation 310
3.3.1 Die Gestaltung kooperativer Aufgabenbereiche 311
3.3.2 Die Bedeutung sozialer Kooperationsstrukturen im Arbeitsprozeß 314
3.3.3 Soziale Integration 318
4 Zusammenfassung 324

IV Software-Entwicklung als kooperative Gestaltungsaufgabe 326

1 Einleitung 326
2 Software-Entwicklung als gemeinsamer Lernprozeß 331
2.1 Lernbeziehung zwischen Entwickler und Anwender 332
2.1.1 Wissensbereiche im Software-Projekt 334
2.1.2 Entwickler und Anwender im Verhältnis zu den Wissensbereichen 338
2.2 Zusammenfassung 341
3 Konflikt und Konsens in der Software-Entwicklung 341
3.1 Konfliktstrukturen von Software-Projekten 342
3.2 Software-Entwicklung im Spannungsfeld antagonistischer Konflikte 346

3.3 Software-Entwicklung im Spannungsfeld konsensual-lösbarer Konflikte 348
4 Kommunikation in der Software-Entwicklung 349
4.1 Kommunikation und Verständigung 350
4.2 Perspektivenverschränkung als Grundlage der Verständigung 351
5 Kooperationskompetenz in der Software-Entwicklung 353
5.1 Kritische Handlungssituation 1: Erarbeitung eines Verständnisses des Arbeitsprozesses 357
5.1.1 Dokumentenanalyse 361
5.1.2 Beobachtungen 362
5.1.3 Befragungsmethoden 363
5.1.4 Besprechungs- und Diskussionsmethoden 365
5.1.5 Zusammenfassung 366
5.2 Kritische Handlungssituation 2: Auseinandersetzung mit Gestaltungsoptionen 366
5.2.1 Dokumentenanalyse 368
5.2.2 Expertengespräche 369
5.2.3 Testverfahren 370
5.2.4 Exkursionen 370
5.2.5 Besprechungs- und Diskussionsmethoden 371
5.3 Kritische Handlungssituation 3: Entwicklung einer Theorie des zukünftigen Arbeitsprozesses 372
5.3.1 Besprechungs- und Diskussionsmethoden 374
5.3.2 Prototyping 375
5.3.3 Testverfahren 377
6 Zusammenfassung 378

V Prozeßsteuerung in der Software-Entwicklung 379

1 Die Steuerung des Software-Entwicklungsprozesses 379
1.1 Die Notwendigkeit der Prozeßsteuerung 379
1.2 Anforderungen an die Prozeßsteuerung 380
2 Produktorientierte Ansätze der Prozeßsteuerung 383
2.1 Modelle als Planungsgrundlage 383
2.2 Das Wasserfallmodell 385

2.3 Statische Prozeßsteuerung über Meilensteine 390
3 Prozeßorientierte Ansätze der Prozeßsteuerung 392
3.1 Das Projektmodell STEPS 393
3.2 Dynamische Prozeßsteuerung durch Referenzlinien 397
3.3 Produkte des Software-Projekts zwischen Software-Konstruktion und Arbeitsgestaltung 399
3.3.1 Theoriebildung 400
3.3.2 Prototyping 403
4 Zusammenfassung 404

Teil C

Qualifizierungsbaustein »Aufgabenverständnis« IT-Fachkräfte im Spannungsfeld zwischen Technikgestaltung und Arbeitsgestaltung

Einleitung 409

AE 1 Einstiegsbaustein »Software Engineering« 413

Vorbemerkung.....413
AS 1 Die aktuelle Situation in der Softwareentwicklung 413
AS 2 Anforderungen an die Entwickler und die Entwicklungsmethoden 414
AS 3 Leitfragen und Aufbau des Qualifizierungsbausteins »Aufgabenverständnis« 414

AE 2 Die Entwicklung der Informationstechnik 416

Vorbemerkung.....416
AE 2.1 Darstellung und Begründung der Ziele und Inhalte der AE 2.....416
AS 1 Einstieg ins Thema: Ein Vergleich zwischen »Damals« und »Heute« 416
AS 2 Das Thema: IT-Fachkräfte im Spannungsfeld zwischen Arbeit und Technik 417
AE 2.2 Der Prozeß der Entwicklung der Informationstechnik 418
Vorbemerkung:.....418
AS 1 Das Instrument zur Untersuchung der Entwicklung.....418
AS 2 Die Ausgangssituation: Der isolierte Großrechner.....419

AS 3 Die Etappe der Durchsetzung der Dialogverarbeitung.....420
AS 4 Arbeit und Technik als »Roter Faden« der Entwicklung.....421

AE 3 Aufgaben von IT-Fachkräften.....422

Vorbemerkung.....422
AE 3.1 Darstellung und Begründung der Ziele und Inhalte der AE 3.....422
AS 1 Zusammenfassung des Standes der Arbeit.....422
AS 2 Überleitung zur Analyse der Aufgaben von IT-Fachkräften.....423
AE 3.2 Das Verhältnis von Arbeit und Technik.....423
AS 1 Vorstellung eines modernen IT-Konzepts.....423
AS 2 Das Verhältnis Arbeit und Technik.....424
AE 3.3 Die Verknüpfung von Arbeitsgestaltung und Technikgestaltung.....427
AS 1 Softwarequalität als Schlüsselanforderung an IT-Fachkräfte.....427
AS 2 Die Ebene der individuellen Arbeitsgestaltung.....428
AS 3 Die Ebene der Gestaltung von Arbeitsbeziehungen.....432
AE 3.4 Die aktuelle Entwicklung der DV und der Aufgaben von IT-Fachkräften.....437
AS 1 Die heutige Situation: Ein Übergangsstadium der Informationsverarbeitung.....437
AS 2 Die Konsequenzen für die Arbeitssituation.....438

AE 4 Aufgabenverständnis und Kompetenzen von IT-Fachkräften.....439

Vorbemerkung.....439
AE 4.1 Darstellung und Begründung der Ziele und Inhalte der AE 4.....440
AS 1 Zusammenfassung des Standes der Arbeit.....440
AS 2 Überleitung zur systematischen Analyse der Anforderungen an IT-Fachkräfte.....440
AE 4.2 Interpretationsrahmen für ein ganzheitliches Verständnis der Aufgaben von IT-Fachkräften.....441
AS 1 Vorstellen des allgemeinen Interpretationsrahmens.....441
AS 2 Der neue Charakter der Technikgestaltung: Technikgestaltung im Arbeitsgestaltungskontext.....442
AS 3 Expertenkooperation.....444

AS 4 Soziale Prozesse 445
AS 5 Technikgestaltung als ganzheitliche Aufgabe 448
AE 4.3 Kompetenzen von IT-Fachkräften 448
Vorbemerkung 448
AS 1 Überblick über wesentliche Kompetenzen von IT-Fachkräften 449
AS 2 Technikgestaltungskompetenz 449
AS 3 Kooperationskompetenz 450
AS 4 Prozeßkompetenz 450
AS 5 Orientierungen für Qualifizierung und Beruf 451

AE 5 Ausgang »Methoden des Software Engineering« 452

AS 1 Anforderungen an den Unterricht 452
AS 2 Umsetzung im Unterricht 452

Literatur 455

Einleitung[1]

Die Qualifizierung von IT-Fachkräften hat eine zentrale gesellschaftspolitische Bedeutung. Das »Weißbuch« der Kommission der Europäischen Gemeinschaften [Europäische Gemeinschaften - Kommission 1993] benennt die Durchsetzung der »Informationsgesellschaft« als strategisches Entwicklungsziel Europas. Dieser Veränderungsprozeß ist in vielfacher Weise mit der Entwicklung der Computertechnologie und damit letztlich mit den Qualifikationen der IT-Fachkräfte verbunden. Sie sind eine »Schlüsselgruppe« für die erfolgreiche Durchsetzung der »Informationsgesellschaft«. Die Realisierung dieses Ziels wird nicht als einfache Implementierung einer neuen technologischen Infrastruktur vonstatten gehen. Entscheidend ist vielmehr, daß es in einem komplexen sozialen Umwälzungsprozeß gelingt, die Industriegesellschaften auf eine neue Grundlage der sozialen und wirtschaftlichen Existenz zu heben. Diese Entwicklung stellt neue Anforderungen an das Berufsverständnis und die Qualifikationen der IT-Fachkräfte.

Die Qualifizierung von IT-Fachkräften hat eine wettbewerbsstrategische Bedeutung. Die Wettbewerbsfähigkeit der Unternehmen beruht mittel- und langfristig auf der Durchsetzung eines neuen Typs der Leistungserstellung. Die Eckpunkte dieses post-tayloristischen Produktionskonzepts sind Flexibilität und Qualität nach außen sowie Integration und Ganzheitlichkeit nach innen. Die Realisierung dieses modernen Produktionskonzepts ist nicht allein als organisatorische Veränderung anzusehen, wie dies bisweilen in Umkehrung der CIM-Euphorie des letzten Jahrzehnts im Umfeld der »lean-management-Diskussion« behauptet wird. Bestimmend für den Erfolg der Reorganisationsbestrebungen wird vielmehr die effiziente Verbindung moderner Organisationsformen mit neuen technologischen Möglichkeiten sein. Dieser Neuorientierungsprozeß bestimmt in vielfacher Weise die Anforderungen an die IT-Fachkräfte. In den Mittelpunkt ihrer Aufgaben rückt ihr Beitrag zur Neugestaltung der betrieblichen Abläufe und Strukturen. Die Qualifizierung der IT-Fachkräfte ist eine unverzichtbare Bedingung dafür, daß sie diese Aufgabe erfolgreich erfüllen können.

---

[1] Die deutsche Sprache läßt eine gefällige geschlechtsneutrale Formulierung nicht zu. Wenn Wert darauf gelegt wird, deutlich zu machen, daß sowohl Frauen als auch Männer gemeint sind, stehen nur verwirrende sprachliche Konstruktionen zur Verfügung. Wir haben uns aus Praktikabilitätserwägungen dafür entschieden, auf diesen sprachlichen Behelf zu verzichten. Wenn wir fortan die gebräuchliche, auf das männliche Geschlecht bezogene Form wählen, beziehen wir hierin Frauen wie Männer gleichermaßen ein.

Die Qualifizierung von IT-Fachkräften hat darüber hinaus eine beschäftigungspolitische Bedeutung. Gegenwärtig ist ein deutlicher Anstieg der Arbeitslosigkeit bei IT-Fachkräften zu verzeichnen. Diesen Anstieg der Arbeitslosigkeit von IT-Fachkräften nur mit dem Verlauf der allgemeinen Konjunktur und teilweise bestehenden Überkapazitäten zu erklären, greift zu kurz. Die Ungleichzeitigkeiten der Entwicklung der einzelnen Marktsegmente der IT-Branche deuten vielmehr auf strukturelle Anpassungsprozesse hin. Die IT-Branche ist gegenwärtig dabei, sich mit neuen Produkten und Absatzstrategien auf grundlegend neue Anforderungen der Kunden einzustellen. Die Anwenderunternehmen ihrerseits vollziehen eine strategische Kehrtwende im Datenverarbeitungs (DV)-Bereich. An der Oberfläche zeigt sich diese Veränderung als Ablösung des Großrechners durch Client-Server-Architekturen. Dieser Prozeß droht die DV-Belegschaften in zwei Segmente mit sehr unterschiedlichen Arbeitsmarktchancen zu teilen: einerseits den neuen Typ der IT-Fachkraft, der die Client-Server-Welt in den Unternehmen aufbaut und mittel- und langfristig gute Beschäftigungschancen hat; andererseits den Mainframe-Spezialisten, der sich in den nächsten zehn Jahren mit den technologischen Altlasten zu begnügen hat und währenddessen seine mittelfristigen Arbeitsmarktchancen immer mehr einbüßt. Die Qualifizierung ist das zentrale Mittel zur Sicherung ihrer Beschäftigungschancen.

Qualifizierung in der Sackgasse?

Die traditionell auf technische Kompetenzen orientierte Aus- und Weiterbildung von IT-Fachkräften steckt dabei in einer Sackgasse. Eine grundlegende Neuorientierung der Qualifizierung von IT-Fachkräften ist unverzichtbar.

Der Wandel der beruflichen Anforderungen für IT-Fachkräfte ist gravierend und keineswegs nur durch den schnellen technischen Fortschritt ausgelöst. Großen Einfluß haben neue IT-Einsatzkonzepte in den Unternehmen. Immer öfter werden heute Informationssysteme für solche Unternehmen entwickelt, die die Strukturen und Abläufe im Unternehmen grundlegend reorganisieren wollen. An die Stelle der spartenorientierten Organisation tritt hier die Orientierung des Unternehmensaufbaus und der Abläufe an den Geschäftsprozessen der Produktgruppen - »Business-Re-Engineering« und »Geschäftsprozeßorientierung« sind wichtige Stichworte dieser Entwicklung. Informationssysteme müssen sich in Folge dieser Veränderung auf die Unterstützung ganzer Vorgangsketten von der Auftragserteilung bis zur Auslieferung und Fakturierung orientieren. Die Orientierung an Geschäftsprozessen geht mit einer Konzentration auf die Unterstützung komplexer, meist wenig strukturierter Sachbearbeitungs- und Managementaufgaben einher. Die technologische Basis dieser Entwicklung sind Client-Server-Architekturen, die die klassischen Großrechner verdrängen.

Die modernen Innovationsprozesse weisen den IT-Fachkräften eine neue Rolle zu und machen Kompetenzen erforderlich, die das traditionelle Qualifikationsprofil eines »Technikers« überschreiten. Gefordert ist heute die Fähigkeit, technisches Know-how in die Innovationsprozesse zur Entwicklung neuartiger Unternehmens- und Arbeitsstrukturen einzubringen. Für IT-Fachkräfte sind damit nicht nur neue Qualifikationen, sondern auch ein neues Berufsverständnis notwendig.

Qualifizierung für die Zukunft

Grundlage eines modernen Qualifizierungskonzepts muß die Orientierung auf ein neues Leitbild sein. Die IT-Fachkraft in kooperativen Entwicklungsprojekten zur Neugestaltung der Arbeitsprozesse in den Unternehmen muß zur neuen Orientierung der Qualifizierung gemacht werden. In den Mittelpunkt der Aus- und Weiterbildung rückt das Ziel, die Qualifikationen zur Gestaltung technischer Systeme mit der Kompetenz zur Kooperation mit den Anwendern und der Fähigkeit zur Bewältigung komplexer sozialer Prozesse zu verschmelzen. Für die Neuorientierung der Qualifizierung stellt das Projekt einen allgemeinen Begründungsrahmen und zwei Qualifizierungsbausteine zur Verfügung, den Qualifizierungsbaustein »Aufgabenverständnis« und das Rahmencurriculum »Moderne Software-Entwicklung«.

In Teil A wird das Qualifizierungsziel »Ganzheitliche Arbeitsgestaltungskompetenz« hergeleitet und die didaktisch-methodische Grundlage zur Vermittlung dieses Qualifizierungsziels in der Aus- und Weiterbildung geschaffen. Wir beginnen hier mit einer Darstellung der Entwicklung des Qualifikationsprofils der IT-Fachkräfte. Diesem Kapitel folgt eine didaktische Fundierung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz«. Wir setzen uns mit den Stärken und Schwächen des Konzepts der Schlüsselqualifikationen auseinander und bemühen uns um eine Reformulierung der traditionellen berufspädagogischen Begriffe der »Fachlichkeit« und »Beruflichkeit« für die Berufsgruppe der IT-Fachkräfte. Aufbauend darauf erfolgt eine theoretische Herleitung der Ansatzpunkte der didaktisch-methodischen Umsetzung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz«. Ausgangspunkt dieses Kapitels ist eine Analyse der einzelnen Bestandteile der beruflichen Handlungskompetenz. Dabei greifen wir in erster Linie auf die psychologische Theorie der Handlungsregulation zurück, die wir mit anderen sozialwissenschaftlichen Theorien verbinden. Die hieraus gewonnenen Ansatzpunkte der beruflichen Handlungskompetenz werden in den folgenden Kapiteln als praktische, didaktische Handlungsbereiche reformuliert. Sie münden in unterrichtspraktische Methoden zum Umgang mit Wissen im Unterricht, zur Vermittlung sozialer

Kompetenzen sowie der Unterstützung einer angemessenen beruflichen Identitätsentwicklung.

In Teil B wird mit dem Rahmencurriculum »Moderne Software-Entwicklung« die inhaltliche Grundlage zur Neuorientierung der Aus- und Weiterbildung von IT-Fachkräften vorgestellt. Das Qualifizierungsziel »Ganzheitliche Arbeitskompetenz« führt zu der Notwendigkeit eines neuen Leitbildes in der Aus- und Weiterbildung. Qualifizierung muß sich in ihren Inhalten auf die Aufgabe der Arbeitsgestaltung beziehen und nach dieser Maßgabe Kompetenzen in den Bereichen »Technikgestaltung«, »Kooperation« und »Prozeßsteuerung« vermitteln. Im zentralen Kapitel des Rahmencurriculums »Software-Entwicklung und Arbeitsgestaltung« werden Ziele und Ansätze der Gestaltung von Arbeit entwickelt und mit Hinweisen zum praktischen Vorgehen verknüpft. Im darauffolgenden Kapitel »Software-Entwicklung als kooperative Gestaltungsaufgabe« wird die Notwendigkeit von Kooperation belegt, das Verhältnis von Anwendern und Entwicklern beschrieben und anhand verallgemeinerter charakteristischer Handlungssituationen im Software-Projekt handlungsorientierend aufbereitet. Das Kapitel »Software-Entwicklung als Prozeß« behandelt Software-Entwicklung als evolutionären und partizipativen Prozeß, dessen Steuerung zu den zentralen Aufgaben der IT-Fachkräfte gehört. Unter dem Fokus der Arbeitsgestaltung werden hier grundlegende Ansätze wie zyklische Vorgehensweisen, Prototyping und partizipative Software-Entwicklung bearbeitet. Diesen, an der Handlungsorientierung ausgerichteten Teilen werden zwei Kapitel vorangestellt, die einen einführenden Charakter haben.

In Teil C wird der Qualifizierungsbaustein »Aufgabenverständnis« vorgestellt. Dieser soll IT-Fachkräfte und Auszubildende dabei unterstützen, sich ein ganzheitliches Verständnis ihrer Aufgaben in modernen Projekten zu erarbeiten. Ein modernes Aufgabenverständnis ist die Voraussetzung dafür, daß bei Qualifizierungsteilnehmern die Bereitschaft zum Erwerb neuer Qualifikationen geweckt wird. Der Qualifizierungsbaustein ist so konzipiert, daß er sich für eine unmittelbare Umsetzung in der Aus- und Weiterbildung eignet. Er gliedert sich in fünf Arbeitseinheiten, die wiederum aus aufeinanderfolgenden Arbeitsschritten bestehen. Im Mittelpunkt stehen die Kapitel »Die Entwicklung der Informationstechnik«, »Aufgaben von IT-Fachkräften« und »Aufgabenverständnis und Kompetenzen von IT-Fachkräften«.

Teil A
Qualifizierungsziel »Ganzheitliche Arbeitsgestaltungskompetenz«

I Zur Entwicklung des Qualifikationsprofils von IT-Fachkräften

Im Forschungs- und Entwicklungsprojekt »Qualifizierungsziel Ganzheitliche Arbeitsgestaltungskompetenz« wurde ein modernes Qualifizierungskonzept für IT-Fachkräfte praxisnah entwickelt und modellhaft erprobt. Dabei wurde davon ausgegangen, daß sich dieses Qualifizierungskonzept in seiner Ausrichtung auf die zukünftigen beruflichen Anforderungen der Berufsgruppe zu beziehen hat. Der Orientierungsrahmen des Qualifizierungskonzepts sollte demnach eine Trendprognose zur zukünftigen Entwicklung der Qualifikationsanforderungen sein. Es stellen sich damit drei zentrale Fragen:

a) Welche Berufsgruppen sind den IT-Fachkräften zuzurechnen?

b) Was macht die Qualifikation von IT-Fachkräften im wesentlichen aus?

c) Wohin entwickelt sich das Qualifikationsprofil der IT-Fachkräfte?

1 Was sind IT-Fachkräfte?

IT-Fachkräfte bilden eine sehr junge Berufsgruppe. In den 40er Jahren entstanden, entwickelte sich ab den sechziger Jahren mit zunehmender Dynamik ein Berufsfeld, das in seiner Struktur sehr eng an die Entwicklung der Computertechnik gebunden ist. Berufsbezeichnungen wie »Lochkartenfachkraft« oder »Bediener von Büromaschinen«, wie sie 1960 für DV-Fachkräfte gebräuchlich waren [vgl. Troll 1982], klingen heute wie aus einer anderen Zeit. Dabei deutet das Befremden, das wir heute bei diesen Berufsbezeichnungen empfinden, an, wie schnell sich das Berufsfeld in den letzten gut drei Jahrzehnten verändert hat.

Die Dynamik dieser Entwicklung führt zu einem disparat strukturierten Berufsfeld. Schon der Begriff »IT-Fachkräfte« wird in der Praxis mehr als Chiffre denn als eingrenzbare Bezeichnung einer Berufsgruppe verwendet. Selbst Spezialisten für Fragen der Klassifizierung der IT-Berufe gelingt es nicht, eine Ordnung in die Berufsgruppe zu bringen. Ein Beleg für diese Behauptung sind die stark divergierenden Schätzungen hinsichtlich der Anzahl der Personen, die dieser Berufsgruppe zugerechnet werden. Die Schätzungen bewegen sich zwischen 350.000 Personen (so die Bundesanstalt für Arbeit für das Jahr 1993 in den alten und neuen Bundesländern) [Computerwoche 12/1994], über 380.000 Personen (so der für die Berufsgruppe zuständige Verantwortliche des Instituts für Arbeitsmarkt- und Berufsforschung) [Dostal 1993, S. 18], bis zu 1.000.000 Personen [vgl. Boß/Roth 1992, S. 41]. Diese großen Unterschiede sind vor allem auf unterschiedliche Zurechnungskriterien zurückzuführen. Einigkeit besteht lediglich

darin, daß die Beschäftigten, die laut Klassifizierung der Berufe in der Berufsordnung 774 als »Datenverarbeitungsfachkräfte« geführt werden, auch als IT-Fachkräfte oder Computerspezialisten anzusehen sind. Sicher ist auch, daß sich noch eine große Anzahl entsprechender Fachkräfte in anderen Berufsordnungen finden läßt, so z.B. in der Berufsordnung 602 »Elektroingenieure« oder der Berufsordnung 612 »Physiker, Physikingenieure, Mathematiker« oder der Berufsordnung 752 »Unternehmensberater, Organisatoren«. [ebd., S. 72]

Diese unübersichtliche Situation ist zwei zentralen Ursachen geschuldet. Die Computertechnologie ist eine »Querschnittstechnologie«, die in sämtlichen Branchen zunehmende Bedeutung gewinnt. In der Praxis vermischen sich die mit dieser Entwicklung entstehenden Berufsgruppen mit den bereits vorhandenen Berufsbildern zu einem undurchsichtigen Gemisch. Dies gilt umso mehr als der schnell wachsende Bedarf an IT-Fachkräften über Jahre hinweg das vorhandene Angebot an entsprechend qualifizierten Fachkräften übertraf und somit bei der Rekrutierung auf andere Berufsgruppen zurückgegriffen werden mußte. Die zweite Ursache liegt darin begründet, daß die Berufsgruppe verhältnismäßig jung und gleichzeitig stark expandierend ist. Mit dieser schnellen Expansion bei gleichzeitigem Fehlen normierender Akteure, die eine systematische Klassifizierung durchsetzen könnten, entsteht ein Gewirr von Berufen.

Diese Entwicklung läßt sich anschaulich am Wachstum der Anzahl der Berufsbezeichnungen nachweisen, die im Bereich der Informationstechnik entstanden sind. Im Jahre 1960 zählte die amtliche Statistik insgesamt 25 Berufsbezeichnungen, im Jahre 1970 waren es bereits 64 Nennungen, im Jahre 1980 waren es 126 Bezeichnungen und für 1989 wurden 243 unterschiedliche Berufsbezeichnungen den DV-Fachkräften zugerechnet. [vgl. Statistisches Bundesamt 1989; zit.n. Boß/Roth 1992, S. 39] Angesichts dieser Unübersichtlichkeit der Berufsgruppe erscheint es unmöglich, eine Trendaussage für die Entwicklung der Qualifikationsanforderungen an IT-Fachkräfte machen zu können.

Dostal hat in dieser mißlichen Lage eine gröbere Unterscheidung vorgeschlagen. Er unterscheidet im Bereich der Informationstechnik Kern-, Misch- und Randberufe, eine Unterscheidung, die in der Praxis mittlerweile gängig ist. [vgl. CDI-Stellenmarktanalysen] Als IT-Fachkräfte wären dieser Unterscheidung folgend die Beschäftigten der IT-Kernberufe einzustufen. Solche sind diejenigen Berufe, in denen sich die Berufsausübung fast vollständig mit Datenverarbeitung befaßt und "die Informationstechnik-Aspekte dominant und ausschließlich" sind. [Dostal 1993, S. 13] Dem entgegengesetzt sind die Beschäftigtengruppen eingestuft, die die Informationstechnik lediglich zur Erfüllung ihrer Arbeitsaufgaben nutzen. Diese werden als Informationstechnik-Randberufe bezeichnet. Zwischen diesen beiden Gruppen siedelt der Autor die Informationstechnik-Misch-

berufe an, die weder den Kern- noch den Randberufen zuzuordnen sind. Sie übernehmen gleichermaßen "Aufgaben aus der Informationstechnik und aus der Anwendung, so daß sich eine klare berufliche Heimat nicht bestimmen läßt." [ebd.] Der Autor verweist darauf, daß jeder dieser Gruppen in der Literatur spezifische IT-Qualifikationen zugeordnet werden. Demnach benötigten die Kernberufe »Systembezogene Informatik«, die Mischberufe »Fachspezifische Informatik« und die Randberufe »Elementare Informatik«. [ebd., S. 14]

Das zentrale Kriterium zur Identifikation von IT-Fachkräften ist dieser Unterscheidung folgend die Beschäftigung mit der Informationstechnik. Bestimmt die Beschäftigung mit dem Computer in hohem Maße die Aufgaben der entsprechenden Berufsgruppe, so ist demnach von einem IT-Kernberuf zu sprechen. Wird die Informationstechnik lediglich am Rande genutzt, so handelt es sich um einen IT-Randberuf. Läßt sich eine klare Unterscheidung diesbezüglich nicht machen, so handelt es sich um einen IT-Mischberuf. Dabei werden keine Unterschiede dahingehend gemacht, ob eine Berufsgruppe einen Computer lediglich als Arbeitsmittel nutzt oder ihn als Arbeitsgegenstand entwickelt und gestaltet. Obwohl diese Differenzierung in der Literatur durchaus zur Unterscheidung der drei Berufsgruppen eingeführt wurde [ebd.], ist sie bei der von Dostal vorgeschlagenen Unterteilung dem Wesen nach nicht enthalten.

Die Informationstechnik erscheint bei Dostal als etwas, das sich systematisch von dem Kontext der Anwendung unterscheidet; Informationstechnik und Anwendung erscheinen gar als polares Gegensatzpaar. Für professionell - bezogen auf die IT-Fachkräfte - hält Dostal dieser Auffassung folgend die Berufsgruppen, die sich ausschließlich mit der Informationstechnik beschäftigen und möglichst nicht mit der Anwendung.[2] Die Gruppe der Mischberufe, die nach seiner Schätzung ca. 3.400.000 Personen umfaßt, hält Dostal daher folgerichtig für ein historisches Übergangsphänomen.[3]

---

[2] Als Entgegnung gegenüber der in Managementkreisen häufig vertretenen Auffassung, daß Generalisten und Hybridqualifikationen bei der Personalauswahl zu bevorzugen seien, wendet er ein: "In der Spezialistentätigkeit wird zwangsläufig ein alleiniger Schwerpunkt gewählt, **entweder die Informatik oder die Anwendung**." [Dostal 1993, S. 23; Herv.d.Verf.]

[3] "Diese Informationstechnik-Mischberufe sind aus der Sicht der Berufsforschung regelrechte »Mißgeburten«, da sie eine mangelhafte Professionalisierung signalisieren und bei jeder Klassifizierung die Strukturen vernebeln, weil die Zuordnung zu diesen Mischberufen dem Wunsch nach eindeutiger Klassifizierung nicht entsprechen kann. Besonders mißlich ist, daß diese Mischberufe von jenen Personen sehr gerne gewählt werden, die ohnehin eine klare Zuordnung nicht vornehmen wollen oder können. Deshalb sollten Mischberufe höchstens ein Übergangsstadium begleiten, aktuelle Lücken im Aufbau spezialisierter Berufe füllen und nach einiger Zeit wieder verschwinden." [Dostal 1993, S. 14]

In den Mittelpunkt der Identifikation der IT-Kernberufe stellt der Autor die Befassung mit dem Computer. Die Computerberufe werden von Twiehaus und Dostal [1990] dementsprechend in einer aufgaben- und tätigkeitsorientierten Klassifizierung entlang folgender fünf Berufsfelder eingeteilt:

- Rechenzentrum,
- Hardware,
- Systemsoftware,
- Anwendersoftware,
- Sonstige DV-Berufe.

Diese fünf Berufsfelder werden wie folgt in Berufe unterteilt:

Rechenzentrum

- Operator,
- Arbeitsvorbereiter,
- Archivar,
- Systembetreuer,
- Hardware- und Softwarespezialisten.

Hardware

- Entwicklungsingenieur,
- Fertigungsfachmann,
- Servicetechniker.

Systemsoftware

- Systemsoftwareentwickler,
- Systemsoftwareprogrammierer,
- Systemsoftwareberater.

Anwendersoftware

- Anwendersoftwareentwickler,
- Anwendersoftwareprogrammierer,
- Anwendersoftwareberater.

Sonstige Computerberufe

- DV-Vertriebsbeauftragter,
- DV-Ausbilder,
- DV-Revisor,
- Datenschutzbeauftragter,
- CAD-Konstrukteur,
- DV-Koordinator,
- DV-Berater. [Twiehaus/Dostal 1990][4]

Die hier vorgeschlagene Unterteilung der Berufsgruppe in insgesamt fünf Untergruppen folgt in ihrer Einteilung im Kern der klassischen Sichtweise des Computersystems, das vereinfachend in Hardware, Systemsoftware und Anwendersoftware unterteilt wird. Dazu wird mit den Rechenzentrumsspezialisten eine Gruppe hinzugenommen, die entlang ihres Einsatzortes zugeordnet wird und eine Residualkategorie vorgeschlagen, der diejenigen zugeordnet werden, die unter den ersten vier Gruppen nicht sinnvoll aufgenommen werden können. Diese Einteilung enthält kein konsistentes Bezugssystem. Einerseits wird eine Unterteilung nach dem Arbeitsgegenstand vorgenommen. Dabei wird eine spezifische Sicht des Computers als »Maschine« mit seinen Komponenten Hardware und Software als Bezugsebene gewählt. Andererseits wird der Einsatzort, nämlich das Rechenzentrum, für die Einteilung gewählt.[5]

---

[4] Über die Größenverhältnisse der einzelnen Berufsgruppen und den jeweiligen Frauenanteil geben die Ergebnisse der Volkszählung von 1987 einen Eindruck, auch wenn die hier gewählten Klassifizierungen nicht mit denen von Twiehaus und Dostal übereinstimmen. Demnach zählen sich 33,5 % der Beschäftigten zur Gruppe der Programmierer und Softwareentwickler (ohne die oben gewählte Unterscheidung von System- und Anwendersoftware nachzuvollziehen). Der Frauenanteil an dieser Gruppe entspricht mit 19 % dem durchschnittlichen Anteil an allem Computerkernberufen. Von den antwortenden Personen rechnen sich 14,2 % zu der Gruppe der Organisatoren und Systemanalytiker. Der Frauenanteil ist hier mit 13 % unterdurchschnittlich hoch. Zu den Rechenzentrums- und Benutzerservice-Fachleuten rechnen sich 18,3 % aller Computerfachkräfte. Der Frauenanteil entspricht hier mit 19 % dem Durchschnitt. Zu den Beratern und Verkäufern im DV-Bereich zählen sich 8,5 % der antwortenden Personen. Der Anteil der Frauen liegt hier bei 16 %. Zu den "sonstigen Computerberufen", einer Gruppe von Berufen, die aufgrund der Angaben nicht den oben genannten vier Gruppen zugeordnet werden konnte, zählen sich 25,5 % der Computerfachleute. Hier ist der Anteil der Frauen mit 35 % am höchsten. [vgl. Dostal 1993, S. 38f.]

[5] Die mangelnde Brauchbarkeit dieser Einteilung für eine Klassifizierung des Berufsfeldes ist an anderer Stelle kritisiert worden [vgl. Boß/Roth 1992] und soll hier nicht weiter verfolgt werden.

Diese Bestimmung der IT-Fachkräfte basiert implizit auf zwei zentralen Prämissen. Erstens wird angenommen, daß der Computer eine Maschine sei, die sich ihrem Wesen nach über die Komponenten Hardware und Software bestimmen läßt.[6] Die Einteilung der Berufsgruppe bestimmt sich so dann nach ihrem Beitrag zur Entwicklung, Herstellung, Bedienung oder Verbreitung dieser Maschine. Solch eine Auffassung impliziert zweitens eine spezifische Sicht des Computers. Als Maschine erscheint er als etwas greifbares und von der Umwelt abgrenzbares. Zwischen der Maschine und der Anwendung wird eine sich ausschließende Dualität gesehen. Diese bestimmt die Sicht auf die Informationstechnik.

In solcher Perspektive erscheint das Entwickeln und Gestalten des Computersystems als ingenieursmäßige Tätigkeit. Die so verstandene Tätigkeit der IT-Fachkraft ist ihrem Wesen nach allein von der Logik der Maschine und den formalen Strukturen des Programms bestimmt und unbeeinflußt von den Eigenheiten des Anwendungskontextes. Selbst Berufe, die mit der Entwicklung von Anwendungssoftware befaßt sind, wie Anwendungsprogrammierer, erscheinen nach dieser Auffassung lediglich mit einem formalisierbaren Extrakt konkreter Anwendungen befaßt, nicht aber mit den konkreten Anwendungen selbst. Zwischen dem Computer und dem Kontext der Anwendung wird eine "feuersichere Brandmauer" [Coy 1992b] errichtet, das Entwickeln und Gestalten des Computers erscheint als Umsetzung vorgegebener formaler Strukturen, so daß die Beschäftigung mit dem Anwendungskontext nicht als Teil der Professionalität der IT-Fachkräfte verstanden wird.

Diese Sicht der IT-Fachkräfte und ihrer wesentlichen Qualifikationen bestimmt bis heute die Aus- und Weiterbildung dieser Berufsgruppe. Ihre Qualifizierung erschöpft sich im Kennenlernen des Computers als Maschine, dem Erlernen des Programmierens als Weiterverarbeitung vorgegebener formaler Strukturen.

Die Auffassung vom Computer als Maschine[7] ist eine traditionelle Sichtweise, die sich - angefangen von der Verwendung des »Universalrechners« für die Berechnung langer Zahlenreihen bis zu den großrechnergestützten Dialogsystemen - als durchaus mit der Realität im Einklang befindlich wähnen konnte. Die DV - auch wenn sie über Terminals in die Anwendungsabteilungen hineinragte -, war räumlich getrennt vom übrigen Geschehen in den Unternehmen. Zwischen der DV-Abteilung und den Anwenderabteilungen bestanden für die Entwicklung von Anwendungssoftware klar geregelte Beziehungen. Die Anwenderabteilungen

---

6 Zur Kritik der Auffassung des Computers als Maschine siehe auch Rammert [1992].

7 Der Wandel der Auffassungen zum Wesen des Computers und des Verhältnisses zwischen Menschen und Computern ist u.a. bei Brauer/Brauer [1992], Oberquelle [1991] und Paetau [1990] anschaulich dargestellt.

hatten klare Vorgaben zu machen, die von der DV-Abteilung entsprechend umgesetzt wurden. Dort, wo die Anwenderabteilungen mit dieser Aufgabenstellung überfordert waren, wurden Scharniergruppen eingerichtet, die zwischen beiden Abteilungen im Sinne konsistenter Anforderungen an die zu entwickelnden Systeme zu vermitteln hatten.

Mittlerweile werden im Umfeld des Bestrebens zur theoretischen Neubestimmung der Informatik[8] neue Sichtweisen auf den Computer und dessen Verhältnis zu den ihn nutzenden Menschen in die Diskussion gebracht. Damit entwikkelt sich auch eine neue Sichtweise dessen, was die IT-Fachkraft ausmacht und welche Qualifikation sie auszeichnet. Brauer und Brauer konstatieren für die Informatik einen Paradigmenwechsel vom "Computer als rechnende Hilfskraft" zu einer neuen Orientierung, die "eine Gruppe von gleichrangigen, selbständigen, einigermaßen intelligenten Akteuren, die bestimmte Aufgaben erledigen und dazu miteinander und mit der Umgebung interagieren" [Brauer/Brauer 1992, S. 15] in den Mittelpunkt der Betrachtung stellt. Die klare Trennung zwischen dem Computer und dem Kontext der Anwendung wird nicht mehr nur in Frage gestellt. Vielmehr tritt die Orientierung der Computergestaltung an der Anwendung und die Frage nach dem jeweils optimalen Verhältnis zwischen formaler und sozialer Welt in den Vordergrund.[9] Die Aufgaben und die Qualifikationen der IT-Fachkräfte bestimmen sich in dieser Diktion nicht mehr allein von der Logik der Maschine her, sondern von deren Wechselverhältnis zum Kontext der Anwendung.[10] Solange die Technik selbst in den Kinderschuhen steckte, ihre Bedie-

---

8 Diese Diskussion läßt sich über die Reihe "Theorie der Informatik", herausgegeben von Coy, erschließen. Einen Überblick bietet Coy u.a. [1992].

9 Nach Fuchs-Kittowski ist das philosophische, theoretische und methodologische Grundproblem der Informatik das Verständnis und die bewußte menschengerechte Gestaltung des Verhältnisses von technischen Automaten und schöpferisch tätigen Menschen oder formalem Modell und nicht-formaler natürlicher und gesellschaftlicher Umwelt. [Fuchs-Kittowski 1992, S. 71]

10 Statt der alleinigen Fokussierung auf Algorithmen und Datenstrukturen fordert die Gesellschaft für Informatik (GI) auch für den Informatikunterricht in der Sekundarstufe II eine stärkere Hinwendung zum Verhältnis des Computers zum Kontext der Anwendung . Unter dem Titel "Veränderte Sichtweisen für den Informatikunterricht" heißt es u.a.: "Bisher ist er (der Informatikunterricht; d.V.) von der traditionellen Auffassung der Informatik als Strukturwissenschaft geprägt. Entsprechend stehen Algorithmen und Datenstrukturen im Mittelpunkt des Unterrichts. (...) Angesichts der Entwicklung der Informatik und der Verbreitung der Informationstechnik ist heute jedoch ein breiteres Spektrum von Problemlösungsansätzen und Anwendungsaspekten weit stärker als bisher zu beachten. (...) Als ordnendes Prinzip wird der Bezug des Menschen zum Computer zugrunde gelegt. Diese Sichtweise ermöglicht die systematische Behandlung von Informatik-Grundlagenwissen zusammen mit Fragen der Sozialverträglichkeit des Werkzeugs Computer, rückt die

nung räumlich getrennt verlief und ihre Entwicklung und Gestaltung systematisch vom Anwendungskontext geschieden war, war das Berufsfeld und dessen Qualifikation von der Logik des Computers als Maschine bestimmt, obwohl eine Unterscheidung des Computers vom Anwendungskontext theoretisch nie - und spätestens seit der Einführung des Dialogsystems auch empirisch nicht mehr haltbar war. In dem Maße, wie der Computer in die Realität integriert wird und seine Gestaltung nicht mehr durch eine »feuersichere Wand« von der Gestaltung der Realität geschieden ist, tritt das Verhältnis von Computer und Anwendungskontext bzw. formalen Strukturen und sozialer Welt [Fuchs-Kittowski 1992] in den Mittelpunkt der Aufgaben von IT-Fachkräften. [11]

Dieser paradigmatische Wechsel bestimmt den Trend der Entwicklung des Berufsfeldes der IT-Fachkräfte und deren Qualifikation. Dennoch gilt diese Entwicklung nicht für alle IT-Berufe gleichermaßen. Die Gruppen, deren Aufgaben auch weiterhin weitgehend von der Logik der Maschine bestimmt sind, also beispielsweise die Rechenzentrumsberufe oder die Spezialisten für bestimmte Hardwareprobleme, werden diesen Trend nicht als Veränderung ihres Aufgabengebietes und der erforderlichen Qualifikationen wahrnehmen, sondern als

---

Computertechnik in einen Zusammenhang mit ihrem Anwender und Gestalter und betont die Wechselwirkung zwischen der Technik und dem Menschen." [Schulz-Zander u.a. 1993, S. 349f.] Auch in der Diskussion um die Ausrichtung des Informatikunterrichts in der Sekundarstufe II finden sich je nach Sichtweise des Computers unterschiedliche Auffassungen über die Leitorientierung. Während die GI vorschlägt, den Unterricht auf das Wechselverhältnis von Mensch und Computer zu orientieren, plädiert beispielsweise Baumann [1990] dafür, "der Kolonisierung der Informatik durch Soziologie und/oder Politologie entschieden entgegen" zu wirken. [vgl. Hauf-Tulodziecki 1993]

[11] Ähnliche Überlegungen spielen bei der Diskussion um die Neubestimmung einer Theorie der Informatik eine wichtige Rolle. Coy formuliert pointiert hinsichtlich der Bestimmung der Disziplin: "Informatik ist somit die Wissenschaft des instrumentellen Gebrauchs der Informationstechnik; einer Sammlung von Instrumenten, mit denen ein soziales Verhältnis, nämlich das der Menschen zu ihrer Arbeit bestimmt wird. Dieses soziale Verhältnis hat mehrere Facetten: Einerseits umfaßt es die Beziehungen arbeitender Menschen zu ihren Arbeitsmitteln und -gegenständen, doch ebenso werden die Beziehungen zwischen den am Arbeitsprozeß beteiligten Menschen und die Beziehungen zwischen den arbeitenden Menschen und den sie determinierenden sozialen und ökonomischen Zwängen durch den Einsatz der Informatik bestimmt." [Coy 1992, S. 18f.] Daher sieht er die Aufgabe der Informatik wie folgt: "Aufgabe der Informatik ist also die Analyse von Arbeitsprozessen und ihre konstruktive, maschinelle Unterstützung. Nicht die Maschine, sondern die Organisation und Gestaltung von Arbeitsplätzen steht als wesentliche Aufgabe im Mittelpunkt der Informatik. Die Gestaltung der Maschinen, der Hardware und der Software ist dieser primären Aufgabe untergeordnet." [ebd., S. 19]

Veränderung der Umfeldbedingungen.[12] Doch diese Beschäftigtengruppen werden nach unserer Auffassung in ihrem Anteil an den IT-Fachkräften zurückgehen, während die anwendungsnahen Berufsgruppen zunehmen. So gesehen sind die IT-Mischberufe auch kein Ausdruck von mangelnder Professionalisierung, wie Dostal meint, sondern ein frühes Indiz für die untrennbare Verbindung von Computer und Anwendungskontext, also geradezu ein »Vorbote« einer modernen IT-Professionalität, die sich vom Wechselverhältnis von Computer und Anwendungskontext herleitet. [13]

2 IT-Fachkräfte an der Schwelle zu einer neuen Qualifikation

Bisher wurde die These vertreten, daß nicht mehr die Logik der Maschine und der formalen Strukturen der alleinige Bezugspunkt für die Entwicklung der IT-Fachkräfte ist. Die Berufsgruppe selbst und ihre Qualifikationen entwickeln sich vielmehr im doppelten Bezugssystem von technischer Entwicklung einerseits und dem Wechselverhältnis des Computers zum Kontext der Anwendung andererseits. Will man demnach Angaben über die Entwicklung des Qualifikationsprofils der IT-Fachkräfte machen, so ist dieses doppelte Bezugssystem genauer zu beleuchten.

Die Qualifikationsanforderungen der IT-Fachkräfte in den Unternehmen befinden sich in einem grundlegenden Wandel. An der Oberfläche macht sich dieser Veränderungsprozeß in einer zunehmenden Vielfalt der geforderten beruflichen Qualifikationen und einer Bedeutungszunahme sozialer Kompetenzen bemerkbar.[14] Ursächlich resultiert dieser Anforderungswandel aus zwei miteinander verbundenen Veränderungsmomenten: der technischen Entwicklung im Bereich

---

[12] Eine solche Veränderung der Umfeldbedingungen erfahren gegenwärtig die Berufsgruppen in den Rechenzentren. Durch "Outsourcing" werden die erforderlichen Rechenleistungen auf spezialisierte Anbieter ausgelagert. Das führt zu einer Reduzierung der Anzahl dieser Beschäftigtengruppen und allgemein gesehen zu einer Umschichtung innerhalb der IT-Fachkräfte dergestalt, daß der Anteil der Rechenzentrumsspezialisten abnimmt, während der der anwendungsnahen IT-Fachkräfte zunimmt.

[13] Ob diese IT-Mischberufe für die Zukunft den Prototyp der modernen IT-Fachkraft darstellen, ist nicht entschieden. Denn ihr Hauptmanko, den Mangel an fundierten technischen Qualifikationen, wiegen sie durch die größere Anwendungsnähe und oft auch durch größere Nähe zur wirtschaftlichen Rationalität der Geschäftsvorgänge auf. Gegenwärtig ist es daher gut möglich, daß sie die Kernberufe in ihrer Bedeutung zurückdrängen. Dies insbesondere dann, wenn die IT-Kernberufe sich die Nähe zur Anwendung nicht erschließen.

[14] Im Vergleich der Jahrgänge der CDI-Stellenmarktanalyse für die Jahre 1991 bis 1993 wird deutlich, daß gerade die Nachfrage nach sozialen Kompetenzen in den Stellenanzeigen besonders stark zunimmt. [CDI-Stellenmarktanalyse 1991, 1992, 1993]

der in Unternehmen eingesetzten Computertechnologie und der Neudefinition des Nutzungskontexts der Informationstechnik im Zusammenhang mit der Herausbildung neuer Unternehmensstrategien.

Der dezentrale Einsatz von Computern, verbunden mit dem Trend zur umfassenden Unterstützung der Arbeit an jedem Arbeitsplatz, führt zu vollkommen neuen Aufgaben und damit zu neuen beruflichen Anforderungen an die IT-Fachkräfte, die weit über die technisch induzierten Veränderungen hinausweisen. Ein Ergebnis dieser Veränderungsprozesse ist die Neudefinition der Qualitätsmaßstäbe für die Arbeit von IT-Fachkräften: technische Kriterien treten in den Hintergrund, einsatzbezogene Kriterien, die aus dem Anwendungskontext des IT-Systems resultieren, gewinnen an Bedeutung. Die Diskussion um Software-Ergonomie ist nur die Spitze des Eisbergs.

Der veränderte Einsatz der Informationstechnik führt weiterhin zu einer veränderten Arbeitsweise der IT-Fachkräfte: Ihre Arbeit wird zunehmend professionalisiert. Systementwicklung verlangt weniger die »spontane Genialität« als vielmehr solide Methodenkenntnisse. Die Arbeit in Projektgruppen und die Kooperation mit Anwendern gehört für viele IT-Fachkräfte bereits zum Arbeitsalltag.

2.1 IT-Gestaltung und veränderter Nutzungskontext

Bis in die sechziger und siebziger Jahre hinein war die spezifische Form des Computereinsatzes die Anwendung eines Großcomputers im Batch-Betrieb in einem Rechenzentrum. Eine strenge Arbeitsteilung und klar definierte Schnittstellen zwischen Fachabteilungen und dem Rechenzentrum prägten das Bild. Der Computer wurde hier zum Berechnen von Massendaten genutzt, wie sie beispielsweise in der Buchhaltung anfielen.

In den achtziger Jahren sah sich die IT-Fachkraft in eine Arbeitsumgebung versetzt, die durch den Großrechner im Terminalbetrieb und den Einsatz von traditionellen Dialoganwendungen gekennzeichnet war. Voraussetzung dafür war, daß DV-Systeme nicht mehr ausschließlich über das Rechenzentrum genutzt, sondern unmittelbar an die Fachabteilungen des Unternehmens angebunden wurden. Dieser Einzug der Computer in die betriebliche Arbeitsrealität ging mit einer veränderten strategischen Nutzung der Datenverarbeitung einher. Das Rationalisierungsziel war nun nicht mehr ausschließlich die bloße Ersetzung von Routinearbeiten durch einen Rechner, sondern die computergestützte Effektivierung betrieblicher Organisations- und Informationsstrukturen.

Mit der betrieblichen Integration der EDV änderten sich auch die Anforderungen an die DV-Fachkräfte. Die Gestaltung der DV-Systeme konnte nun nicht mehr als Laboraufgabe an reine Techniker delegiert werden. Die in der Logik der

Dialoganwendungen angelegte Abbildung betrieblicher Strukturen erforderte zum einen Kenntnisse des Anwendungsfeldes - Stichwort »Fachinformatiker"-, zum anderen setzte die adäquate Abbildung komplexer betrieblicher Strukturen zunehmend den Einsatz ingenieursmäßiger Methoden der Systementwicklung voraus.

Die Computernutzung in den achtziger Jahren ist trotz dieser Hinwendung zu den fachlichen Problemen der Anwendung von ihrem Grundverständnis her in der Tradition der Rechenzentrumskonzeption zu sehen. Die Bedeutung der DV in den Unternehmen bestand weiterhin vorrangig in der Übertragung einzelner, durchaus noch überschaubarer Einzelprozesse auf die »Rechenmaschine«. Zwar machte die Informationstechnik zunehmend einen relevanten Teil des Unternehmens aus, aber nicht im Sinne des integralen Bestandteils des gesamten Unternehmens. Dieser DV-Einsatz konnte noch losgelöst von menschlicher Arbeit geplant und umgesetzt werden. Dagegen vollzieht sich gegenwärtig in den Unternehmen ein grundlegender Umbruch in der Nutzung der IT. In den Unternehmen dringt der PC mehr und mehr in das Arbeitsleben ein. Die genutzte Software dient zunehmend der umfassenden Unterstützung aller Tätigkeiten am Arbeitsplatz. Die starre Großrechnertechnologie wird tendenziell von leistungsfähigen und flexiblen Systemen, seit einiger Zeit verstärkt durch PC-Netze, verdrängt. Die Entwicklung dieser hochflexiblen Systeme erfolgt nicht mehr isoliert am Arbeitsplatz im Rechenzentrum, sondern sie kann nur noch gemeinsam mit dem Benutzer direkt in der Fachabteilung bewältigt werden. Das geschieht zunehmend in Projektgruppen, die aus Mitarbeitern sehr unterschiedlicher Abteilungen zusammengesetzt sind. Begriffe wie Teamfähigkeit, Kontaktfreudigkeit und Projekterfahrung sind deshalb keine Modeerscheinungen, sondern ein deutlicher Indikator dafür, daß die veränderte Aufgabe von IT-Fachkräften in den Unternehmen nicht mehr mit den herkömmlichen technischen Qualifikationen zu bewältigen sind. [Baukrowitz u.a. 1993]

Im Mittelpunkt dieser veränderten Konzepte zur Nutzung des Computers steht die Orientierung des IT-Einsatzes am Leitbild des Computers als Werkzeug und als Kooperations-Infrastruktur. Statt der Ersetzung menschlicher Arbeit durch Automatisierung, gestern und heute die Philosophie des nach der CIM-Euphorie in die Krise geratenen DV-Konzepts, zielt der moderne Einsatz der Informationstechnik auf die Unterstützung der individuellen und kollektiven Arbeitsprozesse.[15]

---

[15] Dieser Wandel in der Nutzung der Informationstechnik wird mit dem empirischen und theoretischen Scheitern der sogenannten "harten KI-These" verstärkt in Informatikerkreisen diskutiert. Besondere Bedeutung hat die Fragestellung für die Bestimmung des Verhältnisses von Mensch und Computer im Kontext der Software-Ergonomie-Diskussion. [vgl. Paetau 1990] Gegenwärtig wird im Umfeld der Debatte zur theoreti-

Dieser paradigmatische Wechsel in der Nutzung der Informationstechnik ist einerseits auf gewachsene technische Möglichkeiten zurückzuführen. Insbesondere das Vordringen der PCs mit hoher Rechenleistung ermöglicht eine flexible Unterstützung heterogener, schwach-strukturierter Arbeitsaufgaben durch geeignete Software-Werkzeuge. Die Dynamik dieses Wechsels, der seinerseits mit einer gravierenden Strukturkrise der IT-Branche verbunden ist, läßt sich aber nur zum Teil auf diese technologische Entwicklung zurückführen. Bestimmend ist vielmehr der sich verändernde Bedarf der Unternehmen.

Viele Unternehmen sind heute auf der Suche nach neuen Unternehmensstrategien. Der zunehmende Konkurrenzdruck auf immer enger werdenden und sich immer rascher wandelnden Märkten verlangt von ihnen ein steigendes Maß an Flexibilität. Dies ist offensichtlich mit den traditionellen Produktionsweisen und betrieblichen Strukturen nicht mehr - bzw. nur in abnehmenden Umfang zu erreichen. Gefordert ist eine umfassende Reorganisation der betrieblichen Abläufe und Strukturen. »Lean production« heißt angesichts dieser Entwicklung das neue Zauberwort für die Wettbewerbsfähigkeit der Unternehmen.[16] Hinter dem inflationären Gebrauch dieses Begriffs steht gegenwärtig weniger ein konkretes Konzept als vielmehr die Suche nach grundlegend neuen Produktionsmodellen. Breite und Intensität der Diskussion lassen einen Konsens dahingehend erkennen, daß eine tiefgreifende Neuorientierung der Unternehmen notwendig ist.

---

schen Neubestimmung verstärkt versucht, eine Bestimmung des aktuellen Leitbildes der Computernutzung zu beschreiben. Brauer/Brauer nähern sich dieser Bestimmung an, in dem sie die Nutzung des Computers am Leitbild der Gruppenarbeit in den Vordergrund stellen. [Brauer/Brauer 1992, S. 15] Oberquelle verweist in Erweiterung der Werkzeugmetapher auf die Perspektive der "Computer-Werkstätten". Ausgehend von der Vorstellung, daß Menschen in modernen Arbeitsprozessen immer mehr wechselnde funktionale Rollen mit unterschiedlichen Aufgaben zu bewältigen haben, sieht er die Bedeutung der Informationstechnik wie folgt: "Jede Rolle kann durch eine spezialisierte Computer-Werkstatt unterstützt werden, in der passende Werkzeuge, Materialien, Schablonen, Behälter u.ä. sowie Kooperationsfunktionen in einer durch den Rollenträger festgelegten räumlichen Ordnung (z.B. sichtbar in einem Fenster auf dem Bildschirm) bereitgehalten werden. Die Kooperation zwischen Rollen (...) kann durch computergestützte direkte Interaktion (gemeinsame Handlungen) oder Verknüpfung der Werkstätten durch gemeinsame Ressourcen erfolgen." [Oberquelle 1991b, S. 55]

[16] Die hypnotische Wirkung des Konzepts "lean-production" scheint von den Autoren der MIT-Studie durchaus intendiert zu sein. Euphorisch heißt es hier beispielsweise: "Schlanke Produktion ist ein überlegener Weg für die Menschen Güter herzustellen. Sie bringt bessere Produkte in größerer Vielfalt zu niedrigeren Kosten hervor. Von gleicher Bedeutung ist, daß sie die Arbeit der Mitarbeiter jeder Ebene, von der Fabrik bis zur Konzernzentrale anspruchsvoller und befriedigender macht. Daraus folgt, daß die ganze Welt die schlanke Produktion übernehmen sollte, und zwar so schnell wie möglich." [Womack u.a. 1991, S. 236]

Diese Diskussion hat eine doppelte Wirkung auf die Entwicklung der IT-Strategien in den Unternehmen.

Einerseits gerät die DV-Abteilung in den Unternehmen zunehmend unter Rechtfertigungsdruck. Dabei sind die Begriffe, mit denen gegen sie zu Felde gezogen wird, der oben genannten Diskussion entlehnt. Es gehört zum guten Ton, im gleichen Atemzug mit der zutreffenden Feststellung, daß die Reorganisation der Arbeit im Mittelpunkt neuer Unternehmensstrategien stehen muß, die falsche anzuführen, daß die Entwicklung der Informationstechnik damit nichts wesentliches zu tun habe. Während in den achtziger Jahren die Wirkung der Computertechnologie für die Wettbewerbsfähigkeit der Unternehmen weit überschätzt wurde - was den DV-Abteilungen einen enormen Machtgewinn und ein relativ kritikfreies Leben beschert hat -, droht man gegenwärtig das Kind mit dem Bade auszuschütten. Daß es dabei auch um eine Verschiebung des Machtgefüges zwischen den Abteilungen geht, liegt auf der Hand.

Andererseits erzeugt die Diskussion um die Reorganisation der Unternehmen einen Sog zur strategischen Neudefinition aller Geschäftsbereiche. Die Verantwortlichen sind gefordert, sich auch bei der Erarbeitung neuer IT-Strategien auf die neuen Erfordernisse einzustellen. Die DV-Abteilungen stehen angesichts dieser Entwicklung in einer vertrackten Lage: entweder sie stellen sich mit neuen IT-Strategien an die Spitze der Bewegung, die an der Reorganisation der Arbeit ansetzen, oder sie werden von den Fachabteilungen in ihrem ureigensten Feld überholt - die neuen Strategien werden dann von dort aus entwickelt.

Unabhängig davon, wie dieser Machtkampf zwischen den Abteilungen entschieden wird - die Realität wird wahrscheinlich beide Entwicklungswege hervorbringen -, verändern die Reorganisationsbestrebungen der Unternehmen die Rahmendaten bei der Entwicklung von IT-Strategien grundlegend. [vgl. Bullinger 1993] Dort, wo versucht wird, die IT-Konzepte mit den neuen strategischen Anforderungen zu verbinden, läßt sich eine Reihe von Veränderungen beschreiben.

Am augenfälligsten für diese Veränderungen ist die Tatsache, daß die Dominanz des Großrechners zunehmend abgelöst wird. Wir erleben aktuell die Verdrängung der Großrechnertechnologie durch verteilte, flexible Rechnerarchitekturen. Dieser Prozeß wird oft mit dem Aussterben der Dinosaurier verglichen; ein Bild, das einerseits die Unumkehrbarkeit des Prozesses demonstrieren und andererseits auf die prinzipielle Unangemessenheit dieser Technologie gegenüber modernen Anforderungen der Informationsverarbeitung hinweisen soll. Dabei sind hier mit der Hardware Programmiersprachen, Software-Philosophien und Einsatz- und Entwicklungsformen betroffen, die einen großen Anteil der

Qualifikationen der IT-Fachkräfte einerseits und des Qualifizierungsangebots der Aus- und Weiterbilder andererseits ausmachen.

Vor wenigen Jahren noch waren Entwicklungen wie Client-Server-Architekturen und Objektorientierung scheinbar Randerscheinungen. Die damals noch relativ geringen Rechenleistungen der Computer engten den Einsatzbereich erheblich ein. Die Entwicklung wurde eher belächelt als ernstgenommen - eine Reaktion die heute nicht mehr verständlich wäre. Innovative Projekte finden v.a. in diesen Bereichen statt und setzen Standards, denen sich Anwenderunternehmen in ihrer Nachfrage immer stärker zuwenden.

Spekulationen darüber, der Bereich der host-orientierten Software-Entwicklung werde noch auf Jahre, wenn nicht Jahrzehnte, Informatiker in erheblichem Umfang beschäftigen, stellen sich heute als Illusion heraus: Statt dessen ist in vielen Unternehmen die Frage der Weiterbeschäftigung der host-orientierten Entwickler zu einem zentralen Personalentwicklungsproblem geworden.

Diese Veränderung der technologischen Basis geht mit der Orientierung des IT-Einsatzes an den Reorganisationsbestrebungen einher. Immer öfter werden heute Informationssysteme für solche Unternehmen entwickelt, die die Strukturen und Abläufe im Unternehmen grundlegend reorganisieren wollen. An die Stelle der spartenorientierten Organisation tritt hier die Orientierung des Unternehmensaufbaus und der Abläufe an den Geschäftsprozessen der Produktgruppen - »Business-Re-Engineering« und »Geschäftsprozeßorientierung« sind wichtige Stichworte dieser Entwicklung. Informationssysteme müssen sich in Folge dieser Veränderung auf die Unterstützung ganzer Vorgangsketten von der Auftragserteilung bis zur Auslieferung und Fakturierung orientieren. Die Orientierung an Geschäftsprozessen geht mit einer Konzentration auf die Unterstützung komplexer, meist wenig strukturierter Sachbearbeitungs- und Managementaufgaben einher. Die technologische Basis dieser Entwicklung sind Client-Server-Architekturen, die die klassischen Großrechner verdrängen.

An der Geschäftsprozeßorientierung läßt sich sehr anschaulich verdeutlichen, welche grundlegenden Veränderungen für die Informatik gegenwärtig zu beobachten sind. Im Zuge der Geschäftsprozeßorientierung werden die Arbeitsplätze aus den starren spartenorientierten Abteilungsstrukturen herausgelöst und in eine durchgängige Vorgangskette von der Auftragserteilung bis zur Auslieferung und Fakturierung eingebunden. Am einzelnen Arbeitsplatz werden einzelne Vorgangsschritte zu ganzheitlichen Aufgabenkomplexen zusammengefaßt. Bisher war der Aufgabenzuschnitt an jedem Arbeitsplatz hochspezialisiert, ein kleiner, überschaubarer Ausschnitt aus dem gesamten Arbeitsablauf. Diesem Ausschnitt entsprach die Software der Fachabteilung: Hochspezialisierte Anwen-

dungen, die nur einen kleinen Teil des Arbeitsablaufes abbilden und weitgehend von anderen Anwendungen isoliert sind. Für die Arbeitsplätze mit ganzheitlichem Aufgabenzuschnitt ist die Unterstützung durch diese alte Software nicht effektiv: sie ist einerseits zu unflexibel und punktuell auf die konkrete Teilaufgabe gerichtet, andererseits ist sie zu wenig transparent.

Die Informationstechnik wird in dieser veränderten Arbeitsorganisation als Kooperations-Infrastruktur und als Werkzeug benötigt, die über den gesamten Arbeitsablauf hinweg Unterstützung bietet. Die Mitarbeiter nutzen den Computer als Werkzeug, um komplexe Informationen flexibel aufzubereiten und die Kommunikation mit anderen Mitarbeitern zu realisieren. Für die Zusammenarbeit von Gruppen, Unternehmensbereichen und ganzen Zulieferketten dient der Computer als Kooperations-Infrastruktur.

Diese modernen DV-Anwendungen sind nicht mehr losgelöst von der Reorganisation der Unternehmensstrukturen zu gestalten. Erfolgreiche Systementwicklungsprozesse erfordern die Integration der technischen Entwicklungsprozesse in die Reorganisation der Arbeitsstrukturen.

2.2 Arbeitsgestaltung als Bezugssystem der IT-Entwicklung

Die Bedeutung der Arbeitsgestaltung als Bezugssystem für die Gestaltung von Computersystemen ist bis heute ein umkämpftes Thema. Zwei Linien stehen sich in der Diskussion unvereinbar gegenüber. Die erste Position negiert die Wirkungen der Neugestaltung eines Computersystems auf die Arbeitsstrukturen oder geht zumindest davon aus, daß der Prozeß der Gestaltung eines Computersystems unabhängig von dessen Wechselwirkung zu der Veränderung der Arbeit durchgeführt werden kann. In dieser, die Systementwicklung in der Praxis dominierenden Diktion - so das Ergebnis einer empirischen Untersuchung [Weltz/Ortmann 1990] - wird die Verantwortung für die Veränderung der Arbeit aus dem Zuständigkeitsbereich der IT-Fachkräfte ausgeklammert. Die Gestaltung der Technik erscheint als rein technischer Prozeß. Brödner [1985] bezeichnet diese Auffassung als "technozentrisch". Die zweite Position verweist auf die Folgewirkungen technischer Veränderung auf die Arbeit und fordert die Anerkennung dieses Wirkungszusammenhangs bei der Gestaltung der Technik. In dieser Diktion ist die Entwicklung von Computersystemen ein Gestaltungsakt, der unvermeidbare Auswirkungen auf die Arbeit hat. Damit negative Folgewirkungen für die Beschäftigten minimiert werden können, sollte diese Arbeitsgestaltungswirkung explizit und nicht implizit stattfinden. In Anlehnung an arbeitspsychologische und industriesoziologische Forschungen zur menschengerechten Gestaltung der Technik wird daher eine Umkehrung des Verhältnisses von Arbeitsgestaltung und Technikgestaltung als notwendig erachtet: Statt die

Arbeitsstrukturen und die Arbeit der Beschäftigten nachträglich an die Erfordernisse des neugestalteten Computersystems anzupassen, wird eine Orientierung der Technikentwicklung an den Prinzipien humaner Arbeitsgestaltung eingefordert. Arbeitsgestaltung vor Technikgestaltung lautet das Credo dieser zweiten, nach Brödner [1985] als "anthropozentrisch" bzw. "arbeitsorientiert" [Ulich 1991] zu bezeichnenden Position.[17]

Während die erste Position die Arbeitsgestaltung als Bezugssystem der Technikgestaltung negiert, ist die Anerkennung dieses Wirkungszusammenhangs für die zweite Auffassung konstitutiv. So unversöhnlich, wie die Auffassungen beider Positionen in dieser Grundfrage sind, basieren sie dennoch hinsichtlich des Verhältnisses von Arbeit und Technik im Gestaltungsprozeß auf vergleichbaren Prämisse. In beiden Fällen wird davon ausgegangen, daß Arbeitsgestaltung und Technikgestaltung unterscheidbare und prinzipiell trennbare Vorgänge sind. Für das Vorgehen im Prozeß wird unterstellt, daß der Gestaltung des Computersystems eine Phase der Definition der Anforderungen vorauszugehen hat, in der festgelegt wird, was später technisch zu realisieren ist. Dabei werden die Anforderungen im Fall der ersten Position unter Ausklammerung der Folgewirkungen auf das Arbeitssystem definiert, während sie im Fall der zweiten Position unter bewußter Orientierung auf die Gestaltungswirkung der Technik hinsichtlich der Arbeit festgelegt werden. Beide Positionen basieren aber trotz aller Unterschiede in ihren Vorstellungen auf der Überlegung, daß der Prozeß der Gestaltung als Aufeinanderfolge von sequentiell verlaufenden Schritten durchgeführt werden kann, und daß die Definition der Anforderungen an das Computersystem systematisch von der technischen Realisierung zu unterscheiden sei. Während also die erste Position die Wechselwirkung von Arbeitsgestaltung und Technikgestaltung im Gestaltungsprozeß negiert, unterstellt die zweite Position die Trennbarkeit von Arbeitsgestaltung und Technikgestaltung im Sinne sequentiell aufeinander aufbauender Schritte des Gestaltungsprozesses.

Damit die Arbeitsgestaltung von der Technikgestaltung zu trennen ist und sequentiell betrieben werden kann, müssen zwei Bedingungen gelten: Erstens muß

---

[17] Diese Sichtweise des Wirkungszusammenhangs von Arbeitsgestaltung und Technikgestaltung ruft gegenwärtig eine Diskussion um die Zuständigkeiten der wissenschaftlichen Disziplinen, die sich mit den beiden Gestaltungsebenen beschäftigen, hervor. Während renommierte Vertreter der Informatik in Anerkennung der arbeitsgestaltenden Wirkung ihres Tuns, die Gestaltung der Arbeitsstrukturen zum neuen Fokus der Disziplin erklären [Coy 1989, S. 257], warnt der Arbeitspsychologe Volpert vor dem Anspruch einer "universalen Gestaltungswissenschaft" und schlägt der Informatik vor, ihren speziellen Beitrag zu einer "Wissenschaft von den Spielräumen des Menschen und dem verantwortlichen Handeln in ihnen" einzubringen. [Volpert 1992a, S. 188]

das Zielsystem[18], in das die neu zu gestaltende Technik eingebracht werden soll, während des Entwicklungsprozesses selbst stabil bleiben. In der Zeitspanne von der Festlegung der Anforderungen an das Computersystem bis zur Implementierung des Systems darf sich das Arbeitssystem nicht - oder nur unwesentlich verändern. Zweitens muß gewährleistet sein, daß der Entwicklungsprozeß selbst eine hohe Zielkonstanz[19] aufweist. Nur so ist zu erreichen, daß das ideale Endprodukt des Prozesses bereits in einem sehr frühen Stadium vollständig und widerspruchsfrei beschrieben werden kann.

Diesen Voraussetzungen folgend werden die Systementwicklungsprozesse auf der Basis eines Phasenmodells mit einem linear-sequentiellen Verlauf der einzelnen Produktionsschritte durchgeführt. Dabei wird entsprechend der Grundauffassung eines sequentiellen Verhältnisses von Technikgestaltung und Arbeitsgestaltung das Verhältnis zwischen der Fachabteilung, für die das System entwickelt werden soll, und der DV-Abteilung, die das technische System zu entwickeln hat, als entsprechend geregeltes Verhältnis angesehen. Von der Fachabteilung wird erwartet, daß sie in der Lage ist, die Anforderungen an das neue Computersystem vollständig und widerspruchsfrei zu bestimmen. Von der DV-Abteilung wird erwartet, daß sie diese Anforderungen in ein fachliches Feinkonzept umsetzt, sie technisch realisiert und danach implementiert. Zwischen beiden Seiten bestehen in dieser Vorstellung klar definierte Schnittstellen.

Die genannten Prämissen, auf denen die Trennung der Technikgestaltung von der Arbeitsgestaltung und ein sequentielles Gestaltungsverhältnis beider Teilprozesse basiert, gelten für moderne Systementwicklungsprozesse nicht mehr. Infolge der grundlegenden Veränderung des Nutzungskontexts moderner Informationstechnik verändert sich auch das Verhältnis von Arbeitsgestaltung und Technikgestaltung im Entwicklungsprozeß grundlegend.

---

[18] Als Zielsystem betrachten wir das (Teil-)Arbeitssystem eines Unternehmens, das durch den Einsatz einer neu gestalteten Informationstechnik verändert werden soll.

[19] Als Ideal von Systementwicklungsprozessen wird angesehen, daß in einem frühen Stadium der Entwicklung das angestrebte Ziel möglichst konkret beschrieben werden kann. Als Zielkonstanz bezeichnen wir den Grad der Beständigkeit des Entwicklungsziels im Verlaufe eines Gestaltungsprozesses. Läßt sich das Ziel bereits früh im Entwicklungsprozeß mit ausreichender Konkretion beschreiben und muß es während der Entwicklung nicht revidiert werden, so ist von hoher Zielkonstanz zu sprechen. Muß demgegenüber das Ziel im Verlaufe des Entwicklungsprozesses revidiert werden und läßt es sich erst spät im Prozeß mit ausreichender Konkretion beschreiben, so sprechen wir von geringer Zielkonstanz. Entwicklungsprozesse weisen dann hohe Zielkonstanz auf, wenn sich der Ausgangszustand und der Endzustand sowie die Operatoren zur Erreichung des Ziels zu Beginn des Prozesses eindeutig bestimmen lassen.

a) Die Veränderung des Zielsystems ist bei modernen Systementwicklungsprozessen geradezu angestrebt. Die Prämisse der Stabilität des Zielsystems gilt nicht mehr. Bei der Entwicklung traditioneller Dialoganwendungen werden Teilaspekte des Arbeitsablaufs - wie bestimmte Rechenoperationen oder hochstandardisierte Tätigkeiten - aus dem konventionellen Ablauf herausgelöst, in formalen Regeln abgebildet, auf den Computer übertragen und in das Arbeitssystem re-integriert. Dabei orientiert sich der Entwickler bei der Erstellung des entsprechenden Anwendungsprogramms an der Vorstellung des Abbildens bereits bestehender Arbeitsabläufe in formalen Strukturen. Moderne Systementwicklungsprozesse sind dagegen nicht mehr an der bloßen Abbildung bestehender Strukturen und Abläufe orientiert, sondern an der Unterstützung der Arbeitsprozesse. Statt hochstandardisierte Abläufe durch die Übertragung auf den Rechner zu effektivieren, wird das Zusammenwirken ganzer Geschäftsprozesse optimiert und es werden schwach strukturierte, komplexe Arbeitsaufgaben unterstützt. Durch die Veränderung der Kommunikationsbeziehung zwischen bisher separierten Abteilungen oder einzelnen Beschäftigten wird deren Kooperationsbeziehung gezielt verändert. Wo vorher die Arbeitsabläufe zweier Abteilungen sequentiell hintereinander gestaffelt verliefen, ergeben sich durch die Umgestaltung der Kommunikationsbeziehung nunmehr reziproke Kooperationsbeziehungen. [Wehrsig/Tacke 1992, S. 224f.] Wo vorher die Arbeitsabläufe zwischen den Beschäftigten einer Abteilung hochgradig arbeitsteilig organisiert waren, werden durch neue IT-Werkzeuge die Voraussetzungen zur Re-Integration der Arbeitsaufgaben - z.B. im Sinne von Rund-um-Sachbearbeitung - geschaffen. In beiden Fällen ist die Veränderung des Zielsystems keine unerwünschte Umfeldveränderung, sondern geradezu das Ziel des Entwicklungsprozesses. Charakteristisch für moderne Systementwicklungsprozesse ist daher, daß mit der Gestaltung der Informationstechnik gestaltend auf das Zielsystem eingewirkt wird.

b) Moderne Systementwicklungsprozesse haben nur noch geringe Zielkonstanz. Die Prämisse hoher Zielkonstanz im Entwicklungsprozeß gilt nicht mehr. Traditionelle Entwicklungsprozesse bauen auf der Prämisse auf, daß der Ausgangszustand und der anzustrebende Endzustand sowie die Operatoren zur Erreichung des Endzustands in einer frühen Phase des Entwicklungsprozesses eindeutig bestimmt werden können. Dieser Prämisse folgend wird der Gestaltungsprozeß als Folge sequentiell aufeinander folgender Entwicklungsphasen organisiert, wobei die jeweils folgende Phase eine Operationalisierung der vorhergehenden ist. Deren Ergebnisse werden hier im Idealfall aber nicht mehr in Frage

gestellt.[20] In modernen Systementwicklungsprozessen läßt sich das Entwicklungsziel erst im Verlaufe einer iterativen Annäherung definieren und erfordert ein zyklisches und evolutionäres Vorgehen. Zu Beginn des Prozesses sind weder der Ist- noch der Sollzustand oder die Mittel und Wege zur Erreichung des Ziels bekannt[21] und müssen erst nach und nach erarbeitet werden. Dieser evolutionäre Prozeß der Erarbeitung des Gestaltungsziels erfordert das permanente Aufeinanderbeziehen der Aspekte des Entwicklungsprozesses. Insbesondere die Wechselwirkungen zwischen Arbeitsgestaltungsaspekten und Technikgestaltungsaspekten ist im Verlauf des Prozesses beständiger Bearbeitungsgegenstand. Während an der Prämisse hoher Zielkonstanz für traditionelle Entwicklungsprozesse mit einer gewissen Berechtigung festgehalten werden konnte, gilt diese für moderne Systementwicklungsprozesse nicht mehr.

c) Moderne Systementwicklungsprozesse erfordern die permanente Zusammenarbeit von Fachabteilungen und DV-Abteilungen sowie von Anwendern und Entwicklern. Aufbauend auf der Prämisse eines stabilen Zielsystems und hoher Zielkonstanz kooperieren Fachabteilung und DV-Abteilung in traditionellen Entwicklungsprozessen im Sinne einer nach klaren Regeln organisierten Schnittstellenbeziehung. Die Vorgaben der Fachabteilung hinsichtlich der fachlichen Erfordernisse im Sinne der Gestaltung der Arbeit werden von der Informatikabteilung in Form eines technischen Systems umgesetzt. Die beiden Aufgaben bauen dabei sequentiell aufeinander auf. Die Verantwortlichen der Fachabteilung haben hier die Funktion, die Anforderungen eindeutig und widerspruchsfrei zu formulieren und die Anwender dieser Abteilungen haben als Informationsquelle für Detailfragen der IT-Fachkräfte zu dienen. Die Forderung nach permanenter Beteiligung der Beschäftigten aus den Fachabteilungen am Entwicklungsprozeß erscheint in traditionellen Entwicklungsprozessen als politische Forderung, zu der je nach Überzeugung unterschiedliche Standpunkte eingenommen werden können. Die Beteiligung der Anwender scheint aber an-

---

[20] Diese Überlegungen liegen auch dem klassischen Software-Engineering zugrunde. Das klassische Phasenmodell, das die schematische, sequentielle Abfolge einzelner Phasen wie Problemanalyse, funktionale Analyse, Entwurf, Programmierung, Funktions- und Leistungsüberprüfung, Installation, Wartung vorsieht, baut nach Hesse auf folgenden Prämissen auf: a) alle für die Entwicklung der Software erforderlichen Vorgaben sind zu Beginn des Projektes vollständig definierbar, b) das hierfür erforderliche Wissen ist zu Beginn des Projektes vorhanden, c) alle Vorgaben des Projektes werden konsensual von allen getragen. [Hesse 1991, S. 18f.]

[21] Einer Klassifizierung von Dörner [1976] folgend beschreiben Sell und Fuchs-Frohnhofen diese, für moderne Systementwicklungsprozesse typische Art der Problemstellung als "dialektisches Problem". [Sell/Fuchs-Frohnhofen 1993]

gesichts der Trennbarkeit von Arbeitsgestaltung und Technikgestaltung nicht prinzipiell notwendig.

Dort, wo moderne IT-Konzepte umgesetzt werden, wird das Verhältnis zwischen der Informatikabteilung und den Fachabteilungen in den Systementwicklungsprojekten im Sinne einer Entwicklungspartnerschaft neu bestimmt. Dabei wird die permanente Einbeziehung der Anwender in die Systementwicklungsprozesse höher gewichtet. Die Einführung der Organisationsform »Projekt« ist als Ausdruck der veränderten Entwicklungsphilosophie zu werten. In Projekten werden alle an einer Innovation beteiligten Funktionsbereiche des Unternehmens unter einem einheitlichen Ziel zusammengebracht. Fachabteilungen und Informatikabteilung bringen hier ihre jeweiligen Kompetenzen in einen gemeinsamen Entwicklungsprozeß ein.[22] Die Beteiligung der Anwender ist in diesen modernen Systementwicklungsprozessen nicht mehr nur eine politische Forderung, sondern immer stärker unverzichtbare Grundlage für erfolgreiche Projekte. Aus der "Nutzerbeteiligung" wird immer mehr eine "kooperative Systementwicklung". [Weltz/Bullinger 1990]

Die grundlegende Veränderung des Nutzungskontexts der IT-Entwicklung und neue technische Möglichkeiten implizieren eine neue Qualität der Durchdringung von Arbeits- und Technikgestaltung. Das sequentielle Gestaltungsverhältnis beider Aspekte des Gestaltungsprozesses wird nunmehr in ein reziprokes Gestaltungsverhältnis transformiert.[23] Diese Veränderung beinhaltet eine verstärkte »Durchmischung« von realer Arbeitswelt und formalem Modell.

---

[22] In unseren Expertengesprächen trafen wir auf ein Unternehmen, das die Systementwicklungs-Philosophie neu konzipiert hatte. Im Mittelpunkt der Neuerungen stand hier die Etablierung des Projekts als verbindlicher Organisationsform für die Abwicklung von Systementwicklungsprozessen, verbunden mit einer stärkeren Einbeziehung der Anwenderabteilungen. Dieses Beispiel illustriert, daß der Anteil der Fachabteilungen am Arbeitsvolumen der Systementwicklungsprojekte große Bedeutung hat: In Projekten soll der durchschnittliche Anteil der Fachabteilungen am Gesamtpersonaleinsatz 50 % betragen, in den Planungsphasen sogar bis zu 80 %.

[23] Trotz dieses veränderten Verhältnisses von Arbeits- und Technikgestaltung ist die Forderung "Arbeitsgestaltung vor Technikgestaltung" prinzipiell richtig. Nur läßt sich diese Forderung nicht mehr durch eine sequentielle Trennung beider Aspekte des Entwicklungsprozesses einlösen. Die prinzipielle Priorität der Arbeitsorientierung gegenüber der Technikorientierung kann nunmehr nicht mehr schematisch verordnet werden, sondern sie läßt sich erst im Laufe eines komplexen Gestaltungsprozesses durch das Handeln der Akteure realisieren. Weil dabei das sequentielle Gestaltungsverhältnis weitgehend hinfällig wird, läßt sich auch die Verantwortung nicht mehr auf unterschiedliche Expertengruppen teilen. Die Gesamtverantwortung für beide Seiten des Entwicklungsprozesses wird vielmehr zur gemeinsamen Verantwortung einer heterogen besetzten Projektgruppe. Daß dieser Gruppe wiederum in der Realität Vorgaben der Gestaltung gemacht werden, die den Gestaltungsspielraum einengen kön-

Die Veränderung des Verhältnisses von Arbeits- und Technikgestaltung wirkt nachhaltig auf das geforderte berufliche Qualifikationsprofil von IT-Fachkräften. Besondere Bedeutung hat dabei die radikale Veränderung des Verhältnisses des Computers zur sozialen Realität, in der er seinen Einsatz findet. Nicht die Größe eines Computers ist entscheidend, nicht die Programmiersprache, in der Software programmiert ist, macht das Neue der Entwicklung aus. Die Realität mit ihrer Komplexität und Mehrdeutigkeit hält Einzug in die Software-Entwicklung.

Bisher konnte man das Verhältnis von Computersystem zur sozialen Realität als genau geregelt ansehen. Hier der Computer mit seinen formalen Strukturen; dort die chaotische, soziale Realität, die man gern aus der Software-Entwicklung heraushalten möchte, und dazwischen eine Schnittstelle, die das Verhältnis zwischen Software und sozialem Umfeld regelt, indem sie dem Menschen die Regeln der Software aufzwingt. Diese Regelung wird von IT-Fachkräften im allgemeinen als sehr praktisch angesehen: Die Entwicklungssituation ist durch eine klar strukturierte Abgrenzung zwischen Software und ihrem Einsatz gekennzeichnet. Computerfachkräfte konnten sich also mehr oder weniger begründet hinter die Grenzen der formalen Welt zurückziehen, die bei aller Komplexität doch sehr viel weniger Probleme aufzuwerfen scheint als reale Verhältnisse.

Mit der Einführung von modernen Softwarearchitekturen und der Orientierung der Systementwicklung an der Leitlinie des »Software-Werkzeugs« und der »Kommunikations-Infrastrukur« wird diese Trennwand durchlöchert. Die Grenzen, in denen formale Strukturen entwickelt werden können, werden immer enger. Komplexer und unschärfer aber werden die Außenbeziehungen der Software-Objekte, die Verhältnisse der Software zum Menschen und zur sozialen Realität.[24]

Damit verbunden ist eine neue Qualität von Verantwortlichkeit der IT-Fachkräfte in ihrer Arbeit. Sie gestalten nicht nur die Computer, sondern das Verhältnis von Software und Mensch. Damit ist ihnen eine Verantwortung übertragen, die aus bisheriger, technisch-orientierter Sicht auf die Arbeit von Informatikern keine Rolle spielt. In neuer Qualität sind nicht nur ökonomische Effizienz, sondern auch

---

nen, soll hier nur erwähnt werden, um nicht einer allzu großen Gestaltungseuphorie das Wort zu reden.

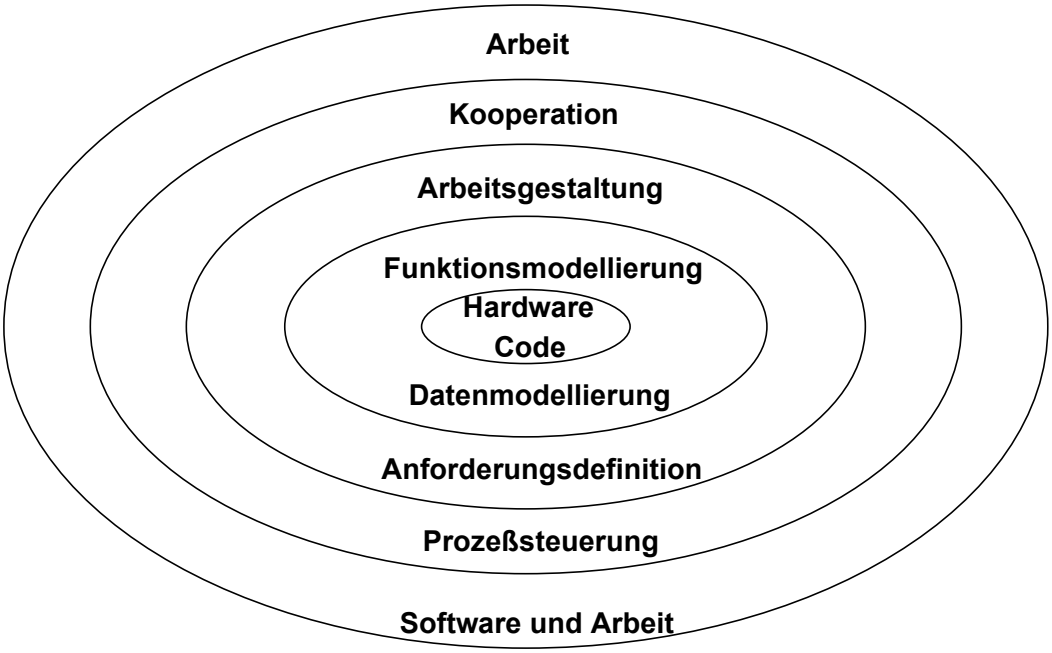
[24] "Die Informatik hat es keineswegs nur mit Formalisiertem zu tun, sondern mit sozialen Prozessen, die in Teilen formalisierbar sind. Die Frage, welches diese Teile sind und wie sie mit den anderen, nicht formalisierbaren zusammenwirken können, ist heute ob man will oder nicht, eine zentrale Fragestellung der Informatik." [Rolf 1993]

die sozialverträgliche Gestaltung von Arbeitsplätzen von ihrem beruflichen Tun berührt.[25]

3 Ganzheitliche Arbeitsgestaltungskompetenz als Qualifikationsprofil der Zukunft

Das veränderte Verhältnis von Computer und sozialer Realität stellt die kommende Herausforderung für Informatiker vor Ort und für ihre Aus- und Weiterbildung dar. IT-Fachkräfte können sich nicht mehr hinter die Grenzen der formalen Welt von Algorithmen und Programmen zurückziehen. Statt dessen wird ihnen eine neue Form von Verantwortlichkeit übertragen, der gerecht zu werden maßgeblich für beruflichen Erfolg und persönliche Zufriedenheit ist. Das Qualifikationsprofil von IT-Fachkräften muß angesichts dieser veränderten beruflichen Anforderungssituation einen neuen Charakter erhalten.

Modernes Qualifikationsprofil von IT-Fachkräften



[25] Im Positionspapier des Fachbereichs »Informatik und Gesellschaft« der Gesellschaft für Informatik wird diese Verantwortungsproblematik beschrieben. [Informatik und Verantwortung 1992]

Folgende Aufzählung beschreibt die notwendigen Kernkompetenzen:

1. IT-Fachkräfte müssen über die Fähigkeit verfügen, Programmcodes zu erzeugen und mit der Hardware umzugehen. Dies gehört zu den klassischen Kompetenzen.
2. Bereits seit einigen Jahren setzt sich die Erkenntnis durch, daß das Programmieren methodisch strukturiert werden muß, um die zunehmende Komplexität der Programme bewältigen zu können. Die Fähigkeit des Programmierens muß also systematisch in die der Datenmodellierung und Funktionsstrukturierung eingebettet sein, um diesem Anspruch gerecht werden zu können.
3. Die Fähigkeit des systematischen Software Engineering zielt ins Leere, wenn sie nicht in einen kompetenten Umgang mit der Bestimmung von Gestaltungsanforderungen an Software eingebettet ist. Da diese nicht offen und eindeutig vorliegen, bedeutet Handlungskompetenz hier, Gestaltungsanforderungen in einem Prozeß der Neugestaltung des jeweiligen Arbeitsprozesses zu entwickeln.
4. Die Arbeitsgestaltung mit ihren Problemen der Anforderungsdefinition, der Umsetzung der Anforderungen in Daten- und Funktionsmodelle, der Programmierung und Implementierung bildet einen komplizierten Erkenntnis- und Entwicklungsprozeß. Evolutionäre Entwicklungsprozesse und die Kooperation mit Anwendern kennzeichnen die Software-Entwicklung, deren Bewältigung und Steuerung besonderer Fähigkeiten bedarf. Arbeitsgestaltung muß deshalb eingebettet sein in die Fähigkeiten zur Kooperation und bewußten Steuerung von Prozessen.
5. Um diese Teilaspekte aufeinander zu beziehen und projektbezogen neu formulieren zu können, benötigen IT-Fachkräfte Orientierungswissen über das Wesen von Arbeit und die Verortung von Software in Arbeitsprozessen. Nur mit diesem Wissen können sie sich den Charakter des konkreten Projekts erarbeiten und sich in allen Anforderungssituationen als kompetent erweisen.

Diese Aufzählung einzelner Kompetenzen verdeutlicht, daß sich das Qualifikationsprofil von IT-Fachkräften grundlegend wandelt. Ihre Qualifikation leitet sich zukünftig nicht mehr vom Computer als Maschine her, sondern vom doppelten Bezug von technischer Entwicklung einerseits und dem Wechselverhältnis des Computers zum Kontext der Anwendung andererseits. Fokussiert man die einzelnen Kompetenzen hinsichtlich der Frage nach den Faktoren, die die Qualifikation von IT-Fachkräften in ihrem Wesen bestimmen, so erhält man drei Bestimmungsmomente des Qualifikationsprofils: Technikgestaltungskompetenz,

Prozeßkompetenz und Kooperationskompetenz. In ihrer Gesamtheit bezeichnen wir diese als »Ganzheitliche Arbeitsgestaltungskompetenz".

Technikgestaltungskompetenz ist das klassische Qualifikationsmerkmal der IT-Fachkräfte. Im Kontext »Arbeitsgestaltung« muß diese Kompetenz neu bewertet und inhaltlich definiert werden. Bestimmend für die IT-Entwicklung werden Kriterien der Arbeitsgestaltung. Im Spannungsfeld zwischen Arbeitsgestaltung und Technikgestaltung besteht der spezifische Beitrag der IT-Fachkräfte darin, den Kontextbezug Arbeit-Technik herzustellen. Im einzelnen bedeutet das:

- Die Technik muß im Kontext von Unternehmensstrategien und Arbeitsgestaltungsprozessen interpretiert und gestaltet werden können.
- Der Technikeinsatz muß im sozialen Zusammenhang interpretiert und gestaltet werden können.
- Die Technik muß im Interessenbezug interpretiert und gestaltet werden können.

Die Arbeitsgestaltung und die Neudefinition von Regulierungsmechanismen sind Gegenstand von Interessenkonflikten und Auseinandersetzungen im Unternehmen. Die Prozesse des Interessenausgleichs und der Auseinandersetzung sind Motor der Innovationsprozesse. Durch die Verknüpfung der IT-Entwicklung mit der Arbeitsgestaltung sind IT-Fachkräfte in diese Prozesse involviert und gestalten sie maßgeblich mit. Das Agieren in sozialen Prozessen rückt daher ins Zentrum ihrer Aufgaben. Um in den komplexen sozialen Prozessen des Innovationsprozesses aktiv und verantwortlich handeln zu können, benötigen sie Prozeßkompetenz. Im einzelnen bedeutet das:

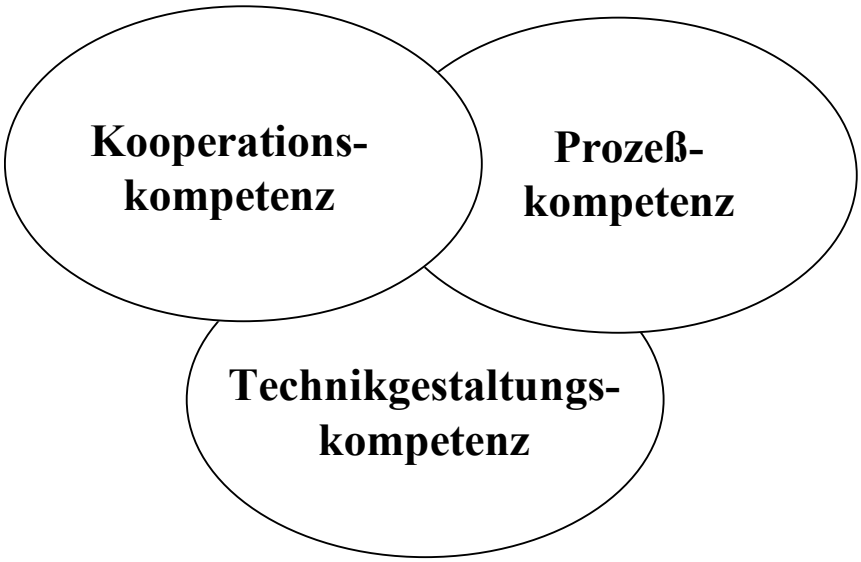
- Die sozialen Prozesse in ihrer Dynamik verstehen zu können.
- Eigene Eingriffsmöglichkeiten und Verantwortung erkennen zu können.
- Interessen und Widersprüche als Motor der Entwicklung verstehen und nutzen zu können.
- Den Entwicklungsprozeß ohne Zielkonstanz regulierend gestalten zu können.
- Zyklizität und Rückkopplung als Methoden der Entwicklung nutzen zu können.

Durch die Verknüpfung von Arbeitsgestaltung und IT-Entwicklung arbeiten IT-Fachkräfte zunehmend in kooperativen Entwicklungszusammenhängen. Kooperation bedeutet, in einer Gruppe gleichberechtigter Experten zusammenzuarbeiten, eigenes Expertenwissen und Entwicklungsleistung als Teile eines übergreifenden Innovationsprozesses einzubringen und sich auf das Expertenwissen

anderer Akteure und deren Entwicklungsleistung in der eigenen Arbeit aktiv zu beziehen. Kooperative Entwicklungsarbeit basiert auf der optimalen Zusammenarbeit und der Leistung eines Teams im Gegensatz zur individuellen Arbeit und Leistung, die in der Berufsgruppe bisher vorherrschend war. Um Synergieeffekte in der Zusammenarbeit der IT-Fachkräfte und der anderen Akteure in komplexen sozialen Strukturen erreichen zu können, wird Kooperationskompetenz zu den notwendigen Qualifikationsmerkmalen der Berufsgruppe gehören. Das bedeutet:

- Die eigene Rolle und Aufgabe im Bezug zu anderen Akteuren im Innovationsprozeß bestimmen zu können.
- Das Expertenwissen und die Entwicklungsleistung anderer Akteure verstehen und bewerten zu können.
- Die eigene Arbeit auf das Expertenwissen und die Entwicklungsleistung der anderen Akteure beziehen zu können.

Ganzheitliche Arbeitsgestaltungskompetenz



Die genannten Kernkompetenzen Technikgestaltungskompetenz, Prozeßkompetenz und Kooperationskompetenz sind in der Realität nicht voneinander zu trennen und bilden eine Einheit. Denn moderne Technikgestaltungskompetenz erfordert zunehmend Kooperations- und Prozeßkompetenz. Ebenso beinhalten die beiden anderen Kernkompetenzen ihrerseits alle anderen. Die Einheit dieser

drei Kernkompetenzen bezeichnen wir als Ganzheitliche Arbeitsgestaltungskompetenz für IT-Fachkräfte. Sie bestimmt zukünftig das Qualifikationsprofil dieser Berufsgruppe.

Diese Feststellung ruft erfahrungsgemäß die Frage hervor, welche Berufsgruppen der IT-Fachkräfte hier gemeint sind. Dabei wird darauf verwiesen, daß die geschilderte Qualifikationsentwicklung bestimmt nicht für die Hardware-Spezialisten gelte oder für die Mehrzahl der Rechenzentrumsberufe. Dem stimmen wir voll zu. Selbstverständlich gilt der von uns konstatierte Trend der Entwicklung des Qualifikationsprofils in Richtung Ganzheitliche Arbeitsgestaltungskompetenz nicht zwingend für alle IT-Berufe und auch nicht in gleichem Maße für jede Berufsgruppe. Unsere These ist vielmehr so zu verstehen, daß sich die IT-Berufe in ihrer Zusammensetzung und das Qualifikationsprofil in seiner Struktur auf ein Kompetenzprofil zubewegen, das wir in der Einheit von Technikgestaltungskompetenz, Kooperationskompetenz und Prozeßkompetenz als Ganzheitliche Arbeitsgestaltungskompetenz bezeichnen.

Dieser Trend zur Herausbildung eines neuen Qualifikationsprofils gilt insbesondere für alle diejenigen IT-Fachkräfte, die in anwendungsnahen Bereichen arbeiten. In den Anwenderunternehmen sind das die Gruppe der Programmierer und die Scharniergruppen zwischen Anwendungsabteilungen und Systementwicklung. Dagegen ist der beschriebene Trend bei den Rechenzentrumsberufen nicht im Sinne einer gravierenden Verschiebung des Qualifikationsprofils zu beobachten, sondern als Rückgang des Anteils dieser Berufe an allen IT-Fachkräften. Der Trend in Richtung Ganzheitliche Arbeitsgestaltungskompetenz muß dabei nicht zwingend zu einer entsprechenden Veränderung der Qualifikationsprofile der angesprochenen Gruppen führen. Dort, wo sich die DV-Abteilungen gegen die Entwicklung und Realisierung moderner IT-Strategien wehren, wird diese Entwicklung für eine gewisse Zeit blockiert werden oder zu einem Bedeutungsverlust der IT-Fachkräfte führen. In einigen Unternehmen zeichnen sich hier Tendenzen ab, neue IT-Konzepte von den Fachabteilungen, die häufig auch die Kompetenz zum Betrieb der PCs haben, zu betreiben. Dies wird dann eher zu einer Ausweitung der sogenannten Mischberufe führen.

Bei den Herstellerunternehmen vollzieht sich gegenwärtig ein regelrechter Strukturbruch, der den beschriebenen Trend in doppelter Weise verstärkt. Trotz ihrer enormen Marktmacht ist es den führenden Hardwareherstellern nicht gelungen, das Vordringen dezentraler Rechnerarchitekturen zu verhindern und die Dominanz der Großrechner aufrechtzuerhalten. Statt dessen versuchen sie jetzt, sich in den expandierenden Geschäftsfeldern »Client-Server-Architekturen« sowie »Software und Service« nachträglich an die Spitze der Bewegung zu stellen. Dies wird in doppelter Weise den Trend in Richtung Ganzheitliche Ar-

beitsgestaltungskompetenz verstärken: Einerseits verlieren die DV-Abteilungen der Anwenderunternehmen ihre Legitimation, die sich bisher gegen neue IT-Konzepte mit dem Argument sperren konnten, daß alle Welt auf Großrechnern oder bestenfalls mit mittlerer Datentechnik arbeitet. Ihre Position ist plötzlich nicht mehr zeitgemäß. Andererseits benötigen die Herstellerunternehmen für die erfolgreiche Realisierung ihrer neuen Geschäftsstrategien verstärkt IT-Fachkräfte, die eine ganzheitliche Qualifikation aufweisen.

Insgesamt spricht vieles dafür, daß sich in den nächsten Jahren der Trend zur Veränderung des geforderten Qualifikationsprofils von IT-Fachkräften in Richtung Ganzheitliche Arbeitsgestaltungskompetenz deutlich verstärken wird.

II Didaktische Fundierung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz«

Vorbemerkung

Der nachfolgende Entwurf einer Didaktik der Ganzheitlichen Arbeitsgestaltungskompetenz beruht auf einer theoriegeleiteten Analyse und zahlreichen Expertengesprächen mit Organisatoren und Lehrenden der Aus- und Weiterbildung von IT-Fachkräften.

Um das nachfolgende didaktische Konzept der Umsetzung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« zu begründen und seine innere logische Kohärenz zu gewährleisten, sind theoretische Überlegungen notwendig, die nur mit dem begrifflichen Werkzeug der sozialwissenschaftlichen Fachsprache zu bewältigen sind.

So notwendig der Gebrauch der Fachsprache für die sozialwissenschaftliche Dokumentation von Forschungsergebnissen auch ist, so problematisch ist sie doch in der vorliegenden Arbeit, insofern in erster Linie die Organisatoren und Lehrenden der Aus- und Weiterbildung die Adressaten des Qualifizierungskonzepts Ganzheitliche Arbeitsgestaltungskompetenz bilden. Wenn wir trotzdem an den Begrifflichkeiten und dem Duktus sozialwissenschaftlicher Arbeiten festhalten, so hat das folgende Gründe:

Die Fachsprache stellt einen Kontext und Verweisungszusammenhang her, den die Alltagssprache unmöglich erbringen kann. Wenn wir beispielsweise von »kommunikativem Handeln« sprechen, nehmen wir dezidiert Bezug auf die Theorie des kommunikativen Handelns von Habermas. Dieser Rückbezug auf sozialwissenschaftliche Theorien bedeutet freilich nicht, daß ohne deren Kenntnis die von uns getroffenen Aussagen nicht zu verstehen sind. Auch der Leser, der kommunikatives Handeln alltagssprachlich mit »miteinander kommunizieren« übersetzt, wird den kontextbezogenen Sinngehalt einer solchen Aussage erfassen. Nur der Problemhorizont, den der Verweis auf die Theorie von Habermas mit einschließt, bleibt dieser alltagssprachlichen Übersetzung verschlossen.

Ein weiterer Grund, warum wir nicht auf die Verwendung der sozialwissenschaftlichen Fachsprache verzichtet haben, ist der Mangel an Alternativen. Die Präsentation von umgangssprachlich formulierten Ergebnissen des Forschungsprojektes hätte zur Folge gehabt, daß der Entstehungs- und Begründungszusammenhang dieser Ergebnisse unweigerlich verlorengegangen wäre,

da dieser notwendig mit fachspezifischen Begrifflichkeiten verbunden ist. Den sozialwissenschaftlichen Begründungszusammenhang in die Fußnoten zu verweisen, erschien uns gleichermaßen als eine wenig geeignete Alternative, da ein solches Vorgehen den Argumentationsgang unserer Arbeit zerrissen hätte.

Der französische Philosoph Deleuze hat den Lesern seiner Bücher den einfachen Rat gegeben: "Findet die Stellen in einem Buch, mit denen ihr etwas anfangen könnt!". So einfach wollen wir es uns mit diesem Problem allerdings nicht machen. Vielmehr erscheint es uns hier notwendig, den Aufbau der folgenden Kapitel kurz zu umreißen, um dem Leser Auskunft über den Abstraktionsgrad und die Funktion einzelner Kapitel zu geben. Das wollen wir deshalb tun, da der Aufbau unserer Arbeit in gewisser Hinsicht dem Weg vom Abstrakten zum Konkreten folgt, der erste Teil also sehr stark theoretisch eingefärbt ist, während der folgende, sich auf diesen beziehende Teil stärker praxisorientiert ist:

In den Kapiteln 1 und 2 des ersten Teils wird das Problematische moderner Qualifizierungskonzepte aus unterschiedlichen Perspektiven herausgearbeitet. Wir setzen uns hierbei sowohl mit den Problemen des nach wie vor aktuellen Konzepts der Schlüsselqualifikationen auseinander, als auch mit der Reformulierung und Bedeutung der traditionellen berufspädagogischen Begriffe der »Fachlichkeit« und »Beruflichkeit«. Diese Kapitel haben insofern einleitende Funktion, als sie grundsätzliche Fragen beruflicher Bildung problematisieren. Diese Problematisierung greift auf berufspädagogische und berufssoziologische Theorien zurück. Im Kapitel 2 haben wir den Versuch unternommen, einige unserer theoriegeleiteten Fragestellungen mit idealtypischen Aussagen von Dozenten der Aus- und Weiterbildung von IT- Fachkräften zu konfrontieren, um die Praxisrelevanz unserer Überlegungen zu verdeutlichen. Trotzdem sind diese Kapitel sehr theoretisch gehalten. Dies hat einen einfachen Grund: Erst mit der Einnahme einer theoretischen Perspektive gewinnen wir den »fremden Blick«, der uns in der alltäglichen Praxis verlorengeht. Erst so werden eingeschliffene Selbstverständlichkeiten hinterfragbar und mögliche Alternativen sichtbar.

Die Kapitel 3 bis 5 des ersten Teils beinhalten die theoretische Herleitung der Ansatzpunkte der didaktisch-methodischen Umsetzung des Qualifizierungsziels Ganzheitliche Arbeitsgestaltung. Ausgangspunkt dieser Kapitel ist eine Analyse der einzelnen Bestandteile der beruflichen Handlungskompetenz. Es geht darum, möglichst exakt diejenigen Faktoren zu bestimmen, die das einzelne Individuum dazu befähigen, beruflich kompetent zu handeln. Dabei greifen wir in erster Linie auf die psychologische Theorie der Handlungsregulation zurück, die wir mit anderen sozialwissenschaftlichen 'Theorien verbinden. Als grundlegende Bestandteile der Handlungskompetenz beschreiben wir den prozessualen Umgang mit Wissen, der auch unter dem u.E. weniger exakten Begriff der Problem-

lösefähigkeit bekannt ist, die für das berufliche Handeln notwendigen sozialen Kompetenzen und die berufliche Identität. Der Begriff der beruflichen Identität ist ein originärer Begriff unserer Arbeit, der sich in der gängigen Diskussion der beruflichen Handlungskompetenz nicht wiederfindet. Wir haben ihn deshalb ausführlich theoretisch begründet. Die den einzelnen Faktoren der beruflichen Handlungskompetenz gewidmeten Kapitel sind jeweils so aufgebaut, daß mit Hilfe sozialwissenschaftlicher Theorien die einzelnen Faktoren zuerst präziser bestimmt und am Schluß der jeweiligen Kapitel auf die spezifische Arbeitssituation von IT-Fachkräften bezogen werden. Die Kapitel 2 bis 5 sind die Kapitel, die sich am engsten an sozialwissenschaftliche Theorien anlehnen. Wenn sich hier die Geduld des nicht sozialwissenschaftlich geschulten Lesers erschöpft, so bitten wir ihn, nicht gleich das ganze Buch aus der Hand zu legen, sondern mit dem folgenden Teil, den Kapiteln des zweiten Teils, fortzufahren.

In den Kapiteln des zweiten Teils werden die theoretisch hergeleiteten Ansatzpunkte der Umsetzung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« auf ihre Bedeutung für das didaktisch-methodische Vorgehen im Unterricht untersucht. Diesen vorausgehend wird im Kapitel 1 des zweiten Teils kurz skizziert, welche organisatorischen Auswirkungen die Neuausrichtung der IT-Ausbildung für die einzelne Ausbildungsinstitution impliziert. Im Kapitel 2 wird ausführlicher dargestellt, welche grundlegenden didaktischen Umorientierungen die Durchsetzung des neuen Qualifizierungsziels auf Seiten der Lehrenden verlangt. Die Kapitel 4 bis 6 bilden den Hauptteil des zweiten Teils unserer didaktisch-methodischen Fundierung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz«. Hier werden gewissermaßen, so hoffen wir, die Früchte der theoretischen Bemühungen geerntet, indem die im ersten Teil gewonnenen Ansatzpunkte der beruflichen Handlungskompetenz als praktische didaktische Handlungsbereiche reformuliert werden. Unterrichtspraktische Methoden zum Umgang mit Wissen im Unterricht, zur Vermittlung sozialer Kompetenzen sowie zur Unterstützung einer angemessenen beruflichen Identitätsentwicklung finden sich in diesen Kapiteln. Mit gewissen Einschränkungen können diese Kapitel auch ohne die theoretischen Herleitungen des ersten Teils gelesen werden.

Wir wollen diese Vorbemerkungen damit beenden, an die Geduld des Lesers zu appellieren, die ihm die Lektüre unserer didaktisch-methodischen Überlegungen abverlangen wird. Die Komplexität des Lehrens von Menschen mit eigenem Willen ist keineswegs geringer als das Programmieren von Maschinen. Ein Teil dieser Komplexität mit all ihren Schwierigkeiten und möglichen Irrtümern findet sich im folgenden wieder.

1 Der besondere Charakter des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« und daraus folgende Schwierigkeiten der Vermittlung

Die Kompetenz zur ganzheitlichen Arbeitsgestaltung als Qualifikationsprofil der Zukunft muß zum Leitbild einer Neuausrichtung der beruflichen IT-Ausbildung werden. Ob die Qualifizierung von IT-Fachkräften sich dieser Qualifikationsnachfrage stellt, wird, wie wir oben gezeigt haben, nicht nur die zukünftige betriebliche Rolle der Ausgebildeten bestimmen, sondern maßgeblich auch deren Arbeitsmarktchancen.

Die Orientierung der IT-Qualifizierung am Leitbild der Ganzheitlichen Arbeitsgestaltungskompetenz erfordert einen Paradigmenwechsel[26] in der Ausbildung, der nicht nur in der Neuformulierung von Ausbildungsinhalten bestehen kann, sondern der auch grundsätzliche didaktisch-methodische Fragestellungen impliziert. Diese didaktisch-methodischen Implikationen sind Thema der folgenden Ausführungen.

Die Umsetzung von Ganzheitlicher Arbeitsgestaltungskompetenz als Paradigma einer neuen Qualifzierungsstrategie zu fordern, ist einfach, und die Zustimmung der Personalentscheider ist einem gewiß. Ohne die Entwicklung von Ansatzpunkten einer Umsetzung des neuen Paradigmas bleibt dieses allerdings nur eine »modische« Erscheinung, die im Alltag der Aus- und Weiterbildung wirkungslos verpufft. Wie wir im folgenden versuchen, deutlich zu machen, kann eine Praxis der neuen Qualifizierungsstrategie nicht auf grundsätzliche theoretische Überlegungen der Pädagogik verzichten. Dieses liegt schon im besonderen Verhältnis des Qualifizierungsziels zur gängigen Unterrichtspraxis begründet. Wir wollen dies kurz verdeutlichen:

Der »normale« Vorgang der Umsetzung neuer Qualifizierungsziele, beispielsweise »die Beherrschung einer bestimmten Programmiersprache«, erfolgt über ihre Integration in den Bildungskanon (Hinzufügen oder Ersetzen) und ihre didaktische Zerlegung in einzelne, linear aufsteigende Lernschritte. Das Qualifizierungsziel »Ganzheitliche Arbeitsgestaltungskompetenz« sperrt sich nun gegen dieses Vorgehensmuster:

---

[26] Es ist modisch geworden, allenthalben einen Paradigmenwechsel zu proklamieren. Wenn wir hier von einem Paradigmenwechsel in der Qualifizierung sprechen, schließen wir uns nicht diesem modischen Trend an, sondern insistieren auf die ursprüngliche Bedeutung des Begriffs. Das heißt: Der Paradigmenwechsel zum Paradigma Ganzheitliche Arbeitsgestaltungskompetenz hat, wie im Verlauf der vorliegenden Arbeit deutlich werden wird, für die Gestaltung der Qualifizierung von IT-Fachkräften einen **programmatischen Charakter** und einen **umfassenden Anspruch**.

- Dem Bildungskanon wird zwar auch Wissen hinzugefügt, aber wesentlich ist, daß er durch eine neue Ausrichtung transformiert wird.
- Die traditionelle didaktische Zerlegung in einzelne, linear aufsteigende Lernschritte wird der Ganzheitlichkeit des neuen Qualifizierungsparadigmas nicht gerecht, das wesentlich durch eine Vernetzung von Qualifikationen unterschiedlicher Ebenen bestimmt ist.

Diese Schwierigkeiten stellen sich nun nicht nur in einem abstrakten Raum pädagogisch-didaktischer Reflexion, sondern vor dem Hintergrund realer Ausbildungsinstitutionen. Unsere Aufgabe lautet demnach, die theoretischen Grundlagen der Vermittlung von Ganzheitlicher Arbeitsgestaltungskompetenz zu entwickeln und auf die Möglichkeiten realer Ausbildungsinstitutionen praktisch zu beziehen. Daß dabei konzeptionelle Kompromisse eingegangen werden müssen, soll nicht nur beklagt werden, zumal die Kompromißlinien »politische« Ansatzpunkte einer Weiterentwicklung von Ausbildungsinstitutionen darstellen.

2 Ansatzpunkte einer Strategie der Vermittlung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz«

2.1 Die notwendige Unterscheidung zwischen Arbeitsprozeß und Arbeitstätigkeit

Bisher haben wir die Ganzheitliche Arbeitsgestaltungskompetenz als notwendige Qualifikation zur Bewältigung der Aufgaben moderner Software-Entwicklung in einem durch Kooperation und Kreativität gekennzeichneten Arbeitsprozeß beschrieben. Diese Perspektive, die Qualifikation aus den »objektiven« Erfordernissen der Arbeit ableitet, kann gewissermaßen als »Außenperspektive« verstanden werden. Diese funktionale, auf den Arbeitsprozeß gerichtete »Außenperspektive« liefert uns eine Matrix, auf die sich die »Innenperspektive« von Qualifikationen beziehen muß. Unter Innenperspektive verstehen wir den Aspekt von Qualifikationen, der die Handlungen der Arbeitenden strukturiert und reguliert, d.h.: Formen der Wahrnehmung, Motivationen, Handlungswissen, das Entscheidungssituationen sichtbar macht und Entscheidungen präferiert, Ausführungswissen und Fertigkeiten, die der Bewältigung einzelner Handlungselemente dienen. Diese Innenperspektive beschreibt den Tätigkeitsaspekt von Qualifikationen als Handlungskompetenz. Auf die Begründung der Relevanz der Unterscheidung dieser zwei Perspektiven und der Notwendigkeit, sie aufeinander zu beziehen, ohne sie aufeinander zu reduzieren, möchten wir an dieser Stelle kurz eingehen, da sie nicht nur eine theoretische Spitzfindigkeit ist, sondern praktische didaktische Folgen hat.

Nehmen wir das einfache Beispiel des Autofahrens. Betrachten wir die dazu erforderlichen Qualifikationen nur aus der »Innenperspektive«, so kann schnell ein Qualifikationsprofil erstellt werden: Die Tätigkeit des Bedienens des Fahrzeugs, das Verstehen der Verkehrsregeln und das Kennen eines Ensembles spezifischer Verhaltensmuster in bestimmten Verkehrssituationen strukturieren und regulieren das Handeln. Betrachten wir dagegen das Autofahren aus einer funktionalen »Außenperspektive«, sehen wir das, was uns die »Innenperspektiv« verstellt hat: es geht um Mobilität. Der Gedanke, nun eine »ganzheitliche Mobilitätskompetenz« - bestehend aus ökologischer Kompetenz, ökonomischer Kompetenz und Sicherheitskompetenz - einzufordern, ist nicht so absurd, wie er auf den ersten Blick zu sein scheint.

Was an diesem Beispiel deutlich wird, ist, daß das Globalziel der Mobilität und seine Implikationen nicht aus der Innenperspektive der Teilqualifikation des Autofahrens gewonnen werden können. Gleichzeitig verhindert oftmals die Fetischisierung des Autos, daß überhaupt eine andere Perspektive als die Innenperspektive entstehen kann. Erst mit dem Einnehmen einer Außenperspektive verfügt die Innenperspektive über ein neues Bezugssystem, das sie selbst zu konstruktiven Veränderungen veranlaßt: Die Teilqualifikation des Autofahrens gewinnt neue Handlungskompetenzen, vom ökologischen und sicheren Fahren bis zum Stehenlassen des Fahrzeugs. Wir wollen hier natürlich keiner platten Analogie zwischen dem Autofahren und der Entwicklung von Software das Wort reden.[27] Dies verbietet alleine schon die unterschiedliche Komplexität der Qualifikationen und der generell anders geartete Charakter des Arbeitsprozesses Software-Entwicklung, der auf die Entwicklung eines durch Unikateigenschaften gekennzeichneten Produktes abzielt. Was allerdings sehr wohl mit diesem Beispiel verdeutlicht werden sollte - und hier gibt es durchaus eine Strukturanalogie - ist, daß die konstruktive Bestimmung von Qualifikationen über die Unterscheidung zwischen einem historisch-gesellschaftlichen Arbeitsprozeß auf der einen Seite und Tätigkeiten, die durch personale Handlungskompetenz strukturiert und reguliert werden, auf der anderen Seite möglich ist.

Dieser Rückbezug der beruflichen Handlungskompetenz auf den Arbeitsprozeß ist für uns nicht zuletzt auch deshalb von zentraler Bedeutung, weil darin der entscheidende Unterschied zwischen dem Konzept der Ganzheitlichen Arbeitsgestaltungskompetenz und dem Konzept der Schlüsselqualifikationen und der

---

[27] Es ist u. E. in erster Linie auch keine Fetischisierung von Programmcodes und Hardware-Eigenschaften, die einem Perspektivewechsel im Bereich der Software-Entwicklung im Weg steht, sondern es ist die technische Dynamik, die gewissermaßen kaum eine »Verschnaufpause« zuläßt, welche oftmals Voraussetzung für den Perspektivewechsel ist.

Handlungsorientierung liegt. Die begriffliche Kongruenz in der Beschreibung der Dimensionen der oben genannten Konzepte spiegelt sich am augenfälligsten in den Begriffen von Kreativität und Kooperation wider. Diese Kongruenz verleitet zu dem Mißverständnis, daß Ganzheitliche Arbeitsgestaltungskompetenz im Grunde nichts anderes als eine Umformulierung des Konzepts der Schlüsselqualifikationen sei. Nicht nur um diesem Mißverständnis vorzubeugen, soll im folgenden das Dilemma des Konzepts der Schlüsselqualifikationen skizziert werden, sondern auch um die positiven Bezugspunkte zu diesem Konzept auszuloten.

2.2 Der Unterschied und die Berührungspunkte des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« und des Konzepts der Schlüsselqualifikationen

2.2.1 Das Dilemma des Konzepts der Schlüsselqualifikationen - eine notwendige Abgrenzung

Als Mertens 1974 das Konzept der Schlüsselqualifikationen entwarf, ging er von Beobachtungen der Veränderungen des Arbeitsmarktes aus. Dabei stellte er eine Dynamik des Wandels der Qualifikationsanforderungen im Beschäftigungssystem fest, die zukunftsorientierte Qualifikationsprognosen fast unmöglich machen würde. Die Ausarbeitung zukunftsorientierter Curriculumskonzepte erschien in dieser Perspektive der Chancenlosigkeit des Hasen im Hase-Igel-Spiel zu gleichen: In der Zeit, bis ein Curriculum entwickelt und in der Praxis umgesetzt wird, hat sich die Qualifikationsanforderung schon so gewandelt, daß das Curriculum dagegen als hoffnungslos veraltet erscheinen muß. Mertens entwickelte daraufhin den genialen Gedanken, ein curriculares Konzept zu entwickeln, das gewissermaßen einem Rollentausch im Hase-Igel-Spiel gleichkommt. Egal, wohin sich die objektiven Qualifikationserfordernisse des Arbeitsprozesses auch bewegen würden, immer würde dem Arbeitenden ein Schlüssel zur Verfügung stehen, mit dem er die konkreten Anforderungen schnell erschließen könnte. Damit bekam "das Konzept der Schlüsselqualifikationen als Flexibilitätsinstrument" [Mertens, zit.n. Lipsmeier 1991, S. 123] eine Richtung, die sich zunehmend von ihrem Ausgangspunkt, der Analyse der Qualifikationsanforderungen im Beschäftigungssystem, entfernte. Mit der Proklamation der Schlüsselqualifikationen entledigte man sich praktisch nicht nur der Mühe, Entwicklungstendenzen der Erwerbsarbeit auf ihre qualifikatorische Bedeutung zu untersuchen, sondern auch eine tiefergreifende Analyse des Arbeitsprozesses erschien als unnötiger Luxus. Trotz seiner großen Verheißungen und anfänglich zahlreich erscheinender Diskussionsbeiträge verschwand das Konzept der Schlüsselqua-

lifikationen, ebenso wie zahlreiche andere Reformvorschläge für die Berufliche Bildung, aus der öffentlich geführten Diskussion.

Als im Rahmen industriesoziologischer Studien Mitte der 80er Jahre eine neue industrielle Revolution ausgerufen wurde, die ganzheitliche Aufgabenzuschnitte in Verwaltung und Produktion versprach und die Notwendigkeit gänzlich neuer Qualifikationsprofile betonte, griffen die Betriebs- und Berufspädagogen stolz in ihre Schublade »Schlüsselqualifikationen« und freuten sich, schon eine Antwort parat zu haben, bevor sie überhaupt gefragt wurden. Die Freude war umso größer, als man im Management einen Bündnispartner fand, der freilich nicht nur die Potenz der »human resources« entdeckte, sondern im Konzept der Schlüsselqualifikationen auch gleichzeitig die Chance erkannte, wie die Risiken des Qualifikationswandels weiterhin beim Arbeitnehmer zu belassen sind. Wir wollen hier nicht weiter gegen die Renaissance der Schlüsselqualifikationen polemisieren.

Vielmehr geht es uns darum, auf ein ernstes Grunddilemma aufmerksam zu machen, das in der Wiederentdeckung der Schlüsselqualifikationen begründet liegt: Das Konzept der Schlüsselqualifikationen wurde in erster Linie mit der Analyse wechselnder Qualifikationsanforderungen in einem dynamisch gewordenen Beschäftigungssystem begründet. Seine Wiederentdeckung war eine Reaktion auf grundlegende Veränderungen im Arbeitsprozeß, die aber, und das ist hier entscheidend, im einzelnen kaum auf ihre qualifikatorische Bedeutung analysiert wurden.[28] Das Ausbleiben einer Neubestimmung von berufsfeldspezifischen Schlüsselqualifikationen[29] anhand einer aufgabenbezogenen Analyse des Arbeitsprozesses hat allerdings nachhaltige Folgen: Das Analyseversäumnis holt das Konzept der Schlüsselqualifikationen spätestens bei der praktischen Umsetzung als eklatantes Defizit wieder ein. So habe, schreibt Zabeck, die Lernpsychologie nachgewiesen, "daß die Steigerung der kognitiven Leistungsfähigkeit im wesentlichen auf den Gegenstand beschränkt bleibt, an dem sie im Lernprozeß erzielt wurde" [Zabeck 1989, S. 80]. Der "Verlust des Fachlichen" [Arnold 1989, S. 87] wird zum allgemeinen Vermittlungsproblem. Der Katalog der

---

[28] Die metaphorische Kraft des Schlüsselbegriffs tut ihr übriges, und es ist Röder beizupflichten, wenn er schreibt: "Beinahe fällt es nicht mehr auf, wenn geäußert wird »Schlüsseltechnologien verlangen Schlüsselqualifikationen«"[Röder 1989, S. 39].

[29] Die "Frage, inwieweit sich möglicherweise berufsfeldspezifische Schlüsselqualifikationen ableiten lassen, die zwischen speziellen Fachkenntnissen und allgemeinen Schlüsselqualifikationen Mertensscher Provenienz anzusiedeln wären" [Badura 1989, S. 24], wurde schon 1975 im Rahmen einer ersten Diskussion des Konzepts von Mertens aufgeworfen [vgl. ebd., S. 31].

Schlüsselqualifikationen wird willkürlich verlängert, je nachdem, wo gerade Qualifikationsbedarf vermutet wird.

Das alles hat zur Folge, daß die Schlüsselqualifikationen im Zusammenhang ihrer praktischen Vermittlung wieder mit fachlichen Qualifikationen verknüpft werden sollen[30]. Diese nachholende Verknüpfung kann aber - obwohl es vorderhand diesen Eindruck erweckt - das oben beschriebene Versäumnis mangelnder Arbeitsprozeßanalyse nicht kompensieren. Um es auf der von den Vertretern des Konzepts selbst ins Spiel gebrachten metaphorischen Ebene auszudrücken: Anstatt erst das Schloß zu analysieren, wird am Design des Schlüssels gefeilt, in der Hoffnung, daß er schon irgendwie passen werde. Die Verknüpfung mit dem Fachlichen bleibt willkürlich, und der Charakter der Schlüsselqualifikation gleicht weiterhin unspezifischen Allgemeinqualifikationen, die auf der "Ebene psychologischer Konstrukte" [Dietrich 1987, S. 27] formuliert werden.

Die prinzipielle Fragwürdigkeit der Konstruktion und Form des Konzepts der Schlüsselqualifikationen soll allerdings nicht darüber hinwegtäuschen, daß mit diesem Konzept Veränderungsprozesse im Bereich der beruflichen Aus- und Weiterbildung initiiert wurden, an denen letztendlich auch wir mit unserem Qualifizierungsziel »Ganzheitliche Arbeitsgestaltungskompetenz« anknüpfen.

2.2.3 Die Wiederbelebung der didaktisch-methodischen Diskussion durch das Konzept der Schlüsselqualifikationen

Mit der Durchsetzung des Konzepts der Schlüsselqualifikationen als Leitidee ging eine Kritik der gängigen didaktischen Grundelemente und Prinzipien der beruflichen Bildung einher. Insbesondere ist eine Verschiebung des berufspädagogischen Interesses zu beobachten. Nachdem jahrzehntelang die Entwicklung neuer Curricula das Hauptthema betriebs- und berufspädagogischer Fragestellungen gebildet hatte, kam es ab den achtziger Jahren zu einer regelrechten "Methodeneuphorie" [Lipsmeier 1991, S. 103]. Sowohl die Kritik gängiger didaktischer Praxis als auch die Entwicklung neuer didaktischer Methoden und Orientierungen bilden Ansatzpunkte zur Vermittlung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz«. Daher ist es u.E. sinnvoll, diese Kritik und ihre produktiven Folgen der Methodenentwicklung und -erprobung an dieser Stelle nachzuzeichnen. Einen ausgezeichneten Überblick zu dieser Pro-

---

[30] Mertens selbst hingegen präferierte eine vom Berufsfeld unabhängige Vermittlung. Er bemühte sich um abstrakte Vehikel und Lerngegenstände, die seine Schlüsselqualifikationen auf ideale Weise transportieren sollten. Zum Erlernen von analytischem Vorgehen empfahl er »Geometrie«, zum Erlernen »kooperativen Vorgehens« Spiele" [Badura 1989, S. 27].

blematik liefern Bojanowski u.a., denen wir im wesentlichen, abgesehen von einigen Anmerkungen, folgen werden. Bojanowski u.a. machen idealtypisch zwei "didaktische Grundelemente" [Bojanowski 1991, S. 86] in dem Verständnis der klassischen Berufsausbildung aus, denen "drei ebenfalls klassische methodische Prinzipien" [ebd. S. 87] entsprechen. Als didaktische Grundelemente erscheinen die »Fachlichkeit« und die »Beruflichkeit«, als methodische Prinzipien die »Vier-Stufen-Methode«, der »Lehrgang« und die »Übungsaufgabe«.

Die Beschränkung der beruflichen Ausbildung auf Fachlichkeit zeigt sich in der ausschließlichen Konzentration auf ein "funktionelles, regelhaftes Wissen und Können um ein »Wenn - Dann«" [ebd. S. 86]. Dieses Wissen soll vom Lernenden nicht nur als Prinzip verstanden werden, sondern »erworben« werden, das heißt so lange geübt werden, bis es »in Fleisch und Blut übergeht«. Das Prinzip der Fachlichkeit schlägt sich in den curricularen Bemühungen nieder, exakt beschriebene berufliche Aufgaben eindeutigem Bearbeitungswissen zuzuordnen.

Die *Beruflichkeit* dient dazu, das Problem zu bewältigen, daß es sachlich weitgehend unbestimmt ist, "welche Fachqualifikationen in welcher Zusammensetzung von jemandem erworben werden" [ebd. S. 86]. Die Beruflichkeit drückt sich in der Ordnungsaufgabe von Berufsbildern aus, die einzelne Fachqualifikationen »bündeln«. Diese "Ordnungsarbeit" [ebd. S. 89] führt zu einer »Schneidung« des Berufes, mit dem weitreichende persönlich-soziale Folgen verbunden sind. Zur Verdeutlichung dieser nichtintendierten, aber um so wichtigeren Folgen wollen wir eine Passage bei Bojanowski u.a. zitieren, zumal sie für unseren weiteren Argumentationsgang von großer Bedeutung ist:

"So kann die Art der Berufsschneidung

- wesentlich die Wahrnehmung der Berufsaufgabe und die Definition der ihr angemessenen Problemlösungen beeinflussen;
- als vorgegebene »Entwicklungsschablone« für die Arbeitenden deren Chancen auf Persönlichkeitsentwicklung und deren Berufsbiographie bestimmen;
- darüber entscheiden, in welchen Kooperations- und Kommunikationsbezügen, in welchen Anweisungs- und Abhängigkeitsstrukturen, mit welchen Selbständigkeitsgraden jemand arbeiten wird" [ebd. S. 89][31]

Alle hier aufgezählten Folgen, die bei der Bündelung von Fachqualifikationen als Beruflichkeit entstehen, werden allerdings bei der praktischen Ordnung von Berufen kaum bedacht. Dies gilt auch und in besonderem Maße für informati-

[31] vgl. ausführlich zur Problematik Beruf - Individuum [Beck u.a. 1980].

onstechnische Ausbildungsberufe, die nur über ein gering ausgebildetes explizites Berufsbild verfügen.

Die diesen didaktischen Grundelementen entsprechenden methodischen Prinzipien prägen weiterhin maßgeblich die berufliche Ausbildung. Diese sind in den Worten von Bojanowski u.a.:

- Die *»Vier-Stufen-Methode«* der Ausbildung, nach der berufliche Lernprozesse in der Folge von Vormachen (Erklären), Nachmachen, Korrigieren und Üben zu verlaufen haben. Diese Methode ist nur auf Einzelqualifikationen anwendbar, die im Sinne eines »einzigen besten Weges« genau festliegen.
- Der *Lehrgang* als systematisch aufgebauter synthetischer Gang des Lernens, dessen innere Logik nicht der Sache, sondern z.B. dem Prinzip »vom Einfachen zum Komplizierten« und »vom Teil zum Ganzen« folgt. Der Lernende kann dabei erst im Rückblick die Teile zu einem sinnvollen Ganzen ordnen.
- Die *Übungsaufgabe* als künstliche Simulation eines Arbeitsvorgangs, dessen Produkt dann als »Edelschrott« weggeworfen wird" [ebd. S. 87].

Die didaktische Kritik der Vertreter des Konzepts der Schlüsselqualifikationen ist leicht nachzuvollziehen. Mit der Fragwürdigkeit der didaktischen Grundelemente der Fachlichkeit und Beruflichkeit verlieren auch die entsprechenden didaktischen Methoden an Wert. Da das Konzept der Schlüsselqualifikationen aber, wie wir oben gezeigt haben, seinen Ausgangspunkt, die Analyse des Arbeitsprozesses, zugunsten einer Orientierung an einem Selbstverständnis als »psychologischem Konstrukt« aufgegeben hat, entsteht notwendig eine eigenartige Fixierung auf die Entwicklung neuer didaktischer Methoden[32], die sich am Subjekt des Lernenden orientiert. Eine kurze Aufzählung soll hier vorerst genügen [vgl. Bojanowski 1991, S. 112ff.; Lipsmeier 1991, S. 108]:

- Leittextmethode
- Projektmethode
- Entdeckendes Lernen
- Qualitätszirkel
- Lernstatt
- Juniorfirma
- PETRA-Modelle (Projekt- und transferorientierte Ausbildung; Siemens)

---

[32] In der berufspädagogischen Diskussion wird oft zu Recht betont, daß diese Methoden prinzipiell schon lange bekannt sind. Aber erst die Diskussion und allgemeine Akzeptanz der Schlüsselqualifikationen hat dazu geführt, daß sie auch praktische Relevanz bekommen haben und dementsprechend verfeinert worden sind.

- TOB-Modelle (Teamorientierte Berufsausbildung; Opel)
- Problemlösungsmethoden.

Auf den hervorgehobenen Stellenwert, den diese Methoden für unser Qualifizierungsziel »Ganzheitliche Arbeitsgestaltungskompetenz« und damit auch für die Verwendung des von uns entwickelten Qualifizierungsbausteins »Aufgabenverständnis« sowie des Rahmencurriculums »Moderne Software-Entwicklung« haben, sei schon hier hingewiesen. Wir werden darauf noch ausführlicher zurückkommen.

Der Beruf- und Fachlichkeit schenken die Vertreter des Konzepts der Schlüsselqualifikationen wenig Beachtung. Beide werden, wenn überhaupt, nur nach den Erfordernissen des psychologischen Konstrukts der Schlüsselqualifikationen reformuliert, d.h. in der Regel ergänzt, ohne den realen Arbeitsprozeß analytisch zu betrachten. Daß diese Ausrichtung des Konzepts der Schlüsselqualifikationen nicht nur von theoretischer Bedeutung ist, hat sich spätestens bei der Neuordnung der Metall- und Elektroberufe gezeigt. Trotz größter Bemühungen konnten die von fast allen hochgehaltenen Schlüsselqualifikationen wenig zur Neubestimmung von Beruf- und Fachlichkeit beitragen:

"Ohne hier ins Detail gehen zu können, sind wir zu der Einschätzung gelangt, daß keine wirklich neue Qualität erreicht wurde, kein Gesamtkonzept zu erkennen ist, das in eine anspruchsvoll zu gestaltende Zukunft weist. Daran kann auch das Kernziel »selbständiges Planen, Durchführen, Kontrollieren« nicht viel ändern, wie es allen Ausbildungszielen als Generalklausel jeweils vorangestellt ist" [Heidegger 1988, S. 151].

Die Ursache für diese traditionelle Formulierung der Fach- und Beruflichkeit liegt sicher darin begründet, daß das "Problem der inhaltlichen Gestaltung der Ausbildungsordnung in enger Anlehnung an die Systematik der analytischen Arbeitsplatzbewertung gelöst wurde" [Gerds 1989, S. 291]. Dem Konzept der Schlüsselqualifikationen ist allerdings vorzuwerfen, daß es dieser traditionellen Sichtweise des Arbeitsprozesses als Ensemble von verschiedenen Tätigkeiten keinen eigenen Analysebegriff entgegenstellen kann.[33]

Damit schließt sich der Kreis unserer Argumentation, deren Ausgangspunkt noch einmal kurz erwähnt werden soll: Die Entwicklung eines modernen Qualifi-

---

[33] Indirekt geben auch Bojanowski u.a. dieses Manko zu, indem sie eine situations- und handlungsorientierte Berufskonzeption vorschlagen, die Qualifikationen aus der Beschreibung berufsfeldspezifischer "Problemsituationen oder Aufgabentypen" [Bojanowski 1991, S. 110] ableitet. "Eine solche situations- und handlungsbezogene Berufskonzeption könnte von vorneherein jedes verkürzte nur-fachliche Mißverständnis der Ausbildung vermeiden."[ebd., S. 111].

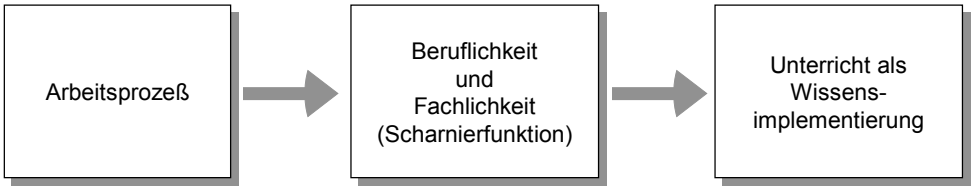
zierungskonzepts muß die von ihm als Qualifizierungsziel herausgestellte berufliche Handlungskompetenz auf eine Analyse des Arbeitsprozesses rückbeziehen. Wenn dies nicht oder auch nur unzulänglich, wie im Falle des Konzepts der Schlüsselqualifikationen, geschieht, kann trotz avanciertester didaktischer Methoden keine erfolgreiche Integration des Konzeptes in die Ausbildungswirklichkeit erwartet werden, da hier gewissermaßen eine praktische Rückbeziehung auf fachliche Qualifikationen stattfinden muß. Das Dilemma des Konzepts der Schlüsselqualifikationen besteht nun gerade darin, daß es den Anspruch auf eine vollwertige Integration in die berufliche Bildung einfordert, aber gleichzeitig jedem Rückbezug auf die Fachlichkeit und Beruflichkeit der beruflichen Ausbildung - aufgrund seiner Ausrichtung als »psychologisches Konstrukt« - aus dem Weg geht, um sein alleiniges Heil in didaktischen Methoden zu suchen.

2.3 Bestimmung der Grundelemente einer Didaktik der Ganzheitlichen Arbeitsgestaltungskompetenz - die Reformulierung von »Fachlichkeit« und »Beruflichkeit«

2.3.1 Die Scharnierfunktion der didaktischen Grundelemente »Beruflichkeit« und »Fachlichkeit« zwischen Arbeitsprozeß und Unterricht

Die Entwicklung eines didaktischen Modells, das sich an dem Qualifizierungsziel »Ganzheitliche Arbeitsgestaltungskompetenz« ausrichtet, muß zuallererst seine didaktischen Grundelemente »Fachlichkeit« und »Beruflichkeit« reflektieren. Zur Neubestimmung dieser didaktischen Grundelemente ist weder die Orientierung an der klassischen Beruf- und Fachlichkeit, noch die Orientierung an dem übergreifenden Konzept der Schlüsselqualifikationen, das diesen Grundelementen nur noch eine Restfunktion zuweist, geeignet. Der Beruf- und Fachlichkeit kommt in unserem Modell praktisch eine zentrale Scharnierfunktion zwischen einer aufgabenorientierten Analyse des Arbeitsprozesses und der Gestaltung des Unterrichts zu. Die Metapher der Scharnierfunktion kann allerdings zu Mißverständnissen führen, wenn sie technologisch verkürzt wird. Dann ist sie nichts weiter als eine einfache Übertragungsfunktion und markiert ein "Verhältnis des einfachen, nicht umkehrbaren Pfeils, der nur eine Richtung und kein Zurück kennt." [Serres 1987, S. 14].

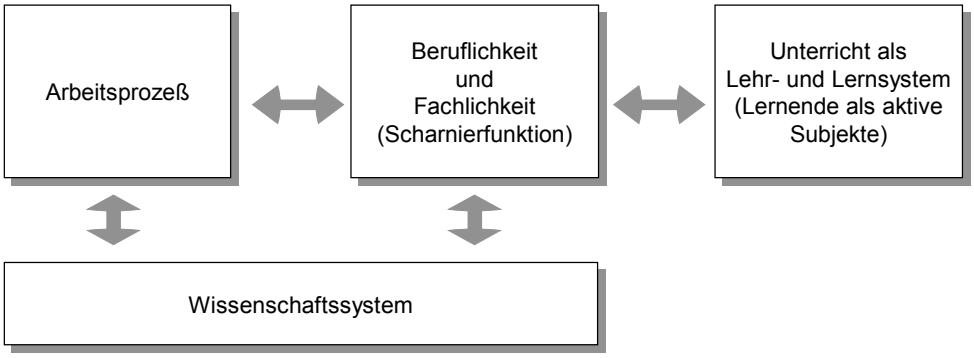
Technologisch verkürzte Sichtweise der Scharnierfunktion der didaktischen Grundelemente Beruflichkeit und Fachlichkeit



Eine nicht technologische verkürzte Darstellung berücksichtigt dagegen sowohl die Subjekthaftigkeit des Lernenden im Unterricht als Lehr- und Lernsystem, als auch den in zahlreichen soziologischen Studien nachgewiesenen Einfluß der professionsgebundenen Beruf- und Fachlichkeit auf den Arbeitsprozeß. Wenn die Profession, wie im Falle der Informatik, auch im Wissenschaftssystem verankert ist[34], erhalten wir dann folgende Abbildung, die die Scharnierfunktion als dialektisches, das heißt offenes und komplexes Verhältnis bestimmt:

[34] Als Wissenschaftssystem sind keineswegs nur Universitäten und Großforschungseinrichtungen zu verstehen, sondern auch die Forschungs- und Entwicklungsabteilungen von Unternehmen. (Von *einem* Wissenschaftssystem kann aufgrund der zunehmenden engen Verflechtung, wie sie sich in den internationalen Vereinigungen ACM und IEEE Computer Society zeigen, gesprochen werden. So hat sich 1992 in den USA z.B. eine »Industry-University Initiative for Software Education« gebildet [vgl. Sledge 1992, VI]. Auf diese Entwicklungstendenzen wollen wir insbesondere auch deshalb hinweisen, weil sich die außeruniversitäre Ausbildung von IT-Fachkräften oftmals intuitiv auf eine Praxis bezieht, die von anderer Seite schon längst als Software-Krise benannt wurde, und an deren Überwindung ein integriertes Wissenschaftssystem arbeitet.) Auf die sich aus diesem Rückbezug ergebenden Probleme der nichtuniversitären Ausbildung von IT-Fachkräften werden wir weiter unten eingehen. Tatsächlich stehen diese beruflichen Ausbildungen vor dem Dilemma, sich auf den »State of the Art« des Wissenschaftssystems beziehen zu müssen und sich aber auch gleichzeitig von der universitären Ausbildung abgrenzen zu müssen. Dabei wird die im Falle der Informatik einzigartige Dynamik des »State of the Art« zum Dauerproblem, da diese nicht nur am Rande des Arbeitsprozesses wirkt, sondern ständig auch die Praxis der Ausbildungsberufe in Frage stellt.

Dialektische Sichtweise der Scharnierfunktion der didaktischen Grundelemente Beruflichkeit und Fachlichkeit



Die zentrale Bedeutung der Neubestimmung der Fachlichkeit und Beruflichkeit ist somit keineswegs nur mit den »objektiven« Erfordernissen des Arbeitsprozesses zu begründen. Vielmehr erscheint es uns schon an dieser Stelle wichtig, auf die »subjektiven Auswirkungen« der Beruflichkeit und Fachlichkeit hinzuweisen. Diese sind insbesondere in berufssoziologischen Untersuchungen nachgewiesen worden. Die Beruflichkeit und Fachlichkeit verleiht dem einzelnen Individuum einen Expertenstatus und in "gewisser Weise Macht über andere" [Beck 1980, S. 220]. Diese Macht zeigt sich in der Verteidigung von Eingriffs- und Gestaltungskompetenzen bezüglich bestimmter Arbeitsinhalte, die selbst wiederum erst durch diese Macht in Form von Experten-Laien-Beziehungen konstituiert werden [vgl. ebd. S. 260]. Die Macht des beruflichen Experten fußt auf einem eigenen »Rationalitätsmonopol« und der realen Abhängigkeit der anderen von seinem Wissen und Können" [ebd. S. 221]. Damit hat sie für das Subjekt allerdings ambivalente Folgen: Einerseits verleiht sie dem Subjekt ein Macht- und Überlegenheitsgefühl, das zur Quelle von Selbstbestätigung und Selbstwertgefühl werden kann, andererseits wird es durch die damit einhergehenden Identifikationsprozesse mit »selbstreferentiellen« Arbeitsinhalten auf eine "berufsspezifische Formung des individuellen Wissens und Könnens" [ebd. S. 214] verpflichtet, deren Marktkonformität krisenhaft verlaufen kann.[35] Auf die sozialwissenschaftliche Diskussion des Zusammenhangs von beruflicher Identität und - zumindest berufswirklichen - Persönlichkeitsstrukturen werden wir weiter unten ausführlich zurückkommen. Wir erwähnen diesen Zusammenhang hier

[35] Auf diese Gefahr für die Informatiker hat jüngst Hartmann hingewiesen [vgl. Hartmann 1993, S. 392ff.].

allerdings schon deshalb, weil er bei der Reformulierung der didaktischen Grundelemente Beruflichkeit und Fachlichkeit von großer Bedeutung ist: Beruflichkeit und Fachlichkeit sind nicht allein aus den objektiven Bedingungen des Arbeitsprozesses reformulierbar, sondern müssen in einer subjektorientierten Didaktik reformuliert werden.

2.3.2 Probleme der Neubestimmung der »Fachlichkeit« und »Beruflichkeit« von IT-Fachkräften

Der naheliegende Versuch, die Fachlichkeit und Beruflichkeit von IT-Fachkräften aus dem Qualifikationsparadigma Ganzheitliche Arbeitsgestaltungskompetenz abzuleiten, stellt uns vor einige Probleme, deren Lösung für die Ausarbeitung einer konsistenten Didaktik notwendig ist. Wir wollen im folgenden diese Probleme skizzieren und im anschließenden Kapitel mögliche Lösungswege entwickeln. Daß diese Lösungswege der Umsetzung des von uns vorgeschlagenen Qualifizierungsparadigmas nicht nur von der Organisation des Unterrichts als Lehr- und Lernsystem, sondern auch von den Bildungsansprüchen der Lernenden abhängig sind, gehört nicht nur zu unseren theoretischen Grundannahmen, sondern ist uns in zahlreichen Gesprächen mit in der Aus- und Weiterbildung Tätigen immer wieder vor Augen geführt worden. Daher werden wir uns diesem Problemfeld auch über idealtypische Aussagen von Praktikern der Aus- und Weiterbildung nähern. Diese Aussagen sind als eine Reaktion auf die Konfrontation mit dem paradigmatischen Anspruch unseres Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« zu verstehen,[36]dessen Dimensionen wir hier nochmals kurz rekapitulieren wollen.

Bedeutung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« für die Neubestimmung der didaktischen Grundelemente Fachlichkeit und Beruflichkeit

Mit dem Anspruch, nicht irgendein Qualifizierungsziel formuliert zu haben, das in den bestehenden Fächerkanon oder in das vorhandene Fach »Informatik und Gesellschaft« integriert werden kann, sondern ein **Leitbild der Qualifizierung** zur Bewältigung der Probleme moderner Software-Entwicklungsprozesse benannt zu haben, wird eine vollständige Veränderung der Fachlichkeit impliziert.

---

[36] Darauf, daß diese Aussagen gleichermaßen die Grenzen sozialwissenschaftlicher Definitionsmacht berühren, soll hier nur am Rande hingewiesen werden: So kann der »Praktiker« das Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« als zentralen und notwendigen Ansatzpunkt einer Neugestaltung der Aus- und Weiterbildung ansehen und gleichzeitig aber auch die Irrelevanz dieses Qualifizierungsziels für seine praktische Unterrichtsgestaltung betonen.

Diese vollständige Veränderung führt nicht nur dazu, daß neue Themen zum Gegenstand der Fachlichkeit werden, sondern auch, daß traditionelle Lehrinhalte uminterpretiert werden müssen. In Anlehnung an Bojanowski u.a. können die traditionellen Kerninhalte der informationstechnischen Ausbildung als ein funktionelles, regelhaftes Wissen und Können um ein »Wenn-Dann« beschrieben werden. Diese Form des Wissens und Könnens erstreckt sich nicht nur auf die technischen Aufgaben, sondern auch auf das Managen des gesamten Entwicklungsprozesses. Eine solche Fachlichkeit kann, wie wir anhand der Analyse des Arbeitsprozesses Software-Entwicklung als sozialem Prozeß gezeigt haben, nicht mehr zum Ausgangspunkt beruflicher Handlungskompetenz werden. Basis einer dem Arbeitsprozeß Software-Entwicklung adäquaten beruflichen Handlungskompetenz kann deshalb u.E. nur eine Fachlichkeit bieten, die über folgende Dimensionen verfügt:

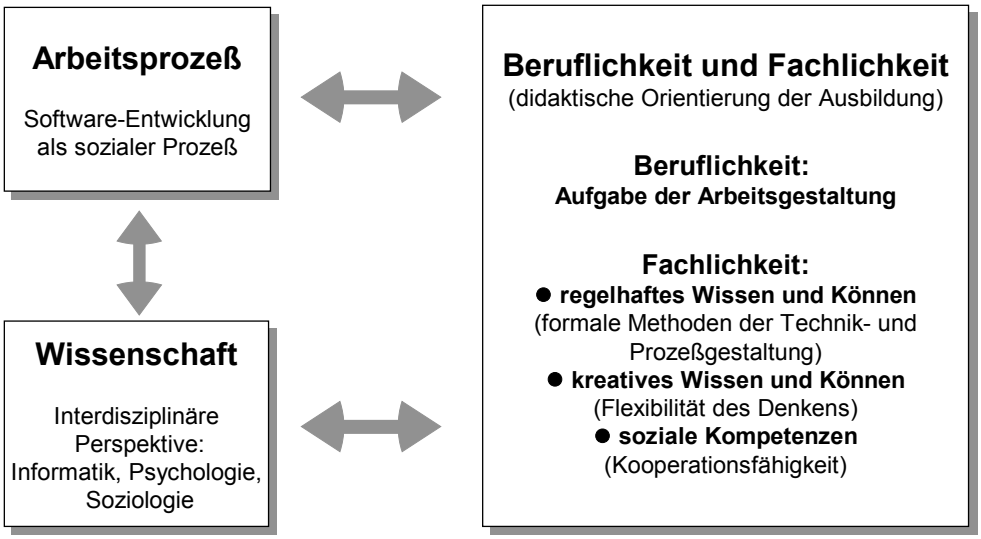
- Regelhaftes Wissen und Können (formale Methoden der Technik- und Prozeßgestaltung),
- Handlungswissen (gestalterisches, kreatives und interpretatives Prozeßregulationswissen).

Gleichzeitig kann diese Fachlichkeit nur dann zum Kern der beruflichen Handlungskompetenz werden, wenn sie durch personengebundene kommunikative, gestalterische und interpretative Fähigkeiten gestützt wird.

Das aus der Analyse des Arbeitsprozesses hervorgehende neue Qualifikationsprofil der IT-Fachkraft haben wir im ersten Kapitel ausführlich dargelegt. Das Qualifikationsprofil Ganzheitliche Arbeitsgestaltungskompetenz beschreibt die zukünftigen Aufgabenbereiche der IT-Fachkräfte in Software-Entwicklungsprozessen. Dieses Qualifikationsprofil bildet die Basis der Entwicklung einer neuen Beruflichkeit.

Alles, was wir hier aufgrund der Analyse des Arbeitsprozesses an Anforderungen für die didaktischen Grundelemente »Fachlichkeit« und »Beruflichkeit« formuliert haben, muß zu ihrer curricularen Neubestimmung herangezogen werden. Diese Analyse ist notwendig, aber nicht hinreichend, um eine Didaktik zu fundieren. Sowohl der Unterricht als Lehr- und Lernsystem als auch die Lernenden als Subjekte bleiben ausgeblendet. Wenn wir unser dialektisches Verständnis der Scharnierfunktion der didaktischen Grundelemente Beruflichkeit und Fachlichkeit als Folie auf die bisherige Darstellung legen, erhalten wir folgende Abbildung.

Bestimmung der didaktischen Grundelemente »Beruflichkeit« und »Fachlichkeit« in ihrem Verhältnis zur Wissenschaft und dem Arbeitsprozeß der Software-Entwicklung



Das Wissenschaftssystem stellt insofern eine eigenständige Einflußgröße dar, als es spezifische Sichtweisen der Informatik generiert und in der Hochschulausbildung auch praktisch realisiert.[37]

Die in zwei Richtungen zeigenden Pfeile verweisen auf den interdependenten Zusammenhang von Arbeitsprozeß und Beruflichkeit + Fachlichkeit, Wissenschaft und Beruflichkeit + Fachlichkeit, sowie Arbeitsprozeß und Wissenschaft. Das heißt für uns: Institutionen der Aus- und Weiterbildung sind aktive Mitgestalter einer bestimmten Beruflichkeit und Fachlichkeit. Die Annahme, daß die Formgestaltung der didaktischen Grundelemente Beruflichkeit und Fachlichkeit einfach aus der »Praxis« des Arbeitsprozesses (»Ausbildung für die Praxis«) resultiert, ist so verbreitet wie falsch. Schon immer hat das Ausbildungssystem

[37] Die Rolle der Wissenschaft wird je nach Standpunkt über-, beziehungsweise unterbewertet. Unabhängig davon, welche Praxisrelevanz bezüglich des Arbeitsprozesses moderner Software-Entwicklung der Wissenschaft nun im einzelnen zugebilligt wird, gilt: Für die Beruflichkeit und Fachlichkeit der IT-Ausbildung hat die Wissenschaft faktisch integrierende Funktion, indem sie die spezifischen Ansprüche der Praxis auf eine allgemeine Fachlichkeit zurückführt.

eine eigenständige Beruflichkeit und Fachlichkeit erzeugt, die als »politischer« Faktor den Arbeitsprozeß mitgestaltet hat.[38] In der Annahme der Verantwortung für eine aktive »politische« Gestaltung der Beruflichkeit und Fachlichkeit durch die Ausbildungsinstitutionen liegen nicht nur die Marktchancen und der berufliche Status der Auszubildenden begründet, sondern sie wird im Extremfall zur eigenen institutionellen Überlebensfrage. Auf unseren Bereich der beruflichen Ausbildung von IT-Fachkräften heißt das zugespitzt: Man kann weiterhin an einer am Programmiercode orientierten Fachlichkeit und Beruflichkeit festhalten und die Organisation der Software-Entwicklung als sozialen Prozeß anderen in der Hoffnung überlassen, daß sich die eigentliche Programmierarbeit vom Prozeß der Software-Entwicklung entkoppeln läßt. Wer aber so argumentiert, muß sich fragen lassen, welche Arbeitsmarktchancen eine solche Fachlichkeit und Beruflichkeit haben, das heißt, wie sich ihre Nische zwischen zunehmend informationstechnisch gebildeten Laien als Endbenutzer und universitär ausgebildeten Informatikern erhalten läßt, und was dies für den Qualifikationsstatus des Ausgebildeten bedeutet. Der pragmatische Hinweis auf augenblicklich noch relativ gute Arbeitsmarktchancen und positive Feedbacks bezüglich der praktischen Verwertbarkeit der Ausbildung aus Unternehmen ist ebenso beruhigend wie trügerisch. Denn er ignoriert:

1. Daß zur Zeit eine grundlegende Reorganisation sowohl von Produktions- als auch Dienstleistungsunternehmen stattfindet, der sich auf Dauer kaum ein Unternehmen entziehen kann; und diese Reorganisation geht mit einer scharfen Kritik an der Beruflichkeit und Fachlichkeit des vorhandenen IT-Personals einher.
2. Daß die Zahl der Absolventen des Studiengangs Informatik an bundesdeutschen Hoch- und Fachhochschulen jährlich um 10-20 Prozent ansteigt, ohne daß eine entsprechende Steigerung bei der Arbeitsmarktnachfrage zu verzeichnen wäre [vgl. Hartmann 1993, S. 398].
3. Daß innerhalb der Diskussion des Hochschulcurriculums für Angewandte Informatik ein paradigmatischer Wechsel der Fach- und Beruflichkeit zunehmend gefordert wird[39]. Dieser tritt mit dem Anspruch auf, einen neuen

---

[38] So hat sich das Facharbeiterwesen in Deutschland aufgrund seiner Eigenkonstruktion von Beruflichkeit und Fachlichkeit gegen »objektive« Tendenzen in der Industriearbeit halten und Formen handwerklicher Produktion selbst in der tayloristisch organisierten Fabrik bewahren können [vgl. Sabel/Piore 1985]. Ähnlich eindrucksvoll liest sich die Geschichte der Ingenieure und der Ingenieursausbildung [vgl. Neef 1982].

[39] vgl. [Wendt 1993; Ludewig 1992; Maaß 1993, Volpert 1993].

»State of the Art« zu setzen, auf den sich auch die berufliche IT-Ausbildung beziehen muß.

Die hier angesprochenen Probleme verweisen auf die zweite Bezugsebene der Fach- und Beruflichkeit, wie wir sie in unserem Modell »Dialektisches Verständnis der Scharnierfunktion der didaktischen Grundelemente Beruflichkeit und Fachlichkeit« weiter oben entwickelt haben: dem Unterricht als Lehr- und Lernsystem einerseits und die Lernenden als Subjekte andererseits. Die folgende Betrachtung ist zwar theoretisch inspiriert, aber ihre inhaltliche Ausgestaltung beruht maßgeblich auf Gesprächen mit zahlreichen Praktikern der Aus- und Weiterbildung. Diesem Umstand wollen wir dadurch gerecht werden, daß wir das folgende Kapitel anhand von idealtypischen Aussagen von Praktikern strukturieren, denen wir unsere - als solche gekennzeichnete - Interpretationen gegenüberstellen. Dadurch bekommt das Kapitel eine quasi-dialogische Gestalt.

Der Unterricht als Lern- und Lehrsystem und die lernenden Subjekte oder die Frage, welche Fachlichkeit und Beruflichkeit sich berufliche Aus- und Weiterbildung zur Aufgabe machen kann

Sobald Beruflichkeit und Fachlichkeit zum Gegenstand beruflicher Ausbildung wird, gewinnt die »Perspektive der Ausbildung« an Bedeutung. Diese »Perspektive der Ausbildung« beinhaltet keineswegs nur methodische Fragen der Stoffvermittlung, auf die wir weiter unten eingehen werden, sondern sie reflektiert die Beruflichkeit und Fachlichkeit hinsichtlich ihrer institutionellen organisatorischen Möglichkeiten und der Ansprüche der Auszubildenden. Die organisatorischen Möglichkeiten einer Ausbildungsinstitution sind abhängig von deren Organisationskultur, Flexibilität und Reformfähigkeit. Diese Faktoren eröffnen einen mehr oder weniger großen Handlungsspielraum, der sich zwischen einzelnen Ausbildungsinstitutionen beträchtlich unterscheiden kann. Dieser relative Handlungsspielraum wird oftmals durch die Behauptung, daß absolute Handlungsgrenzen existieren würden, legitimiert. Hier endet dann die Definitionsmacht des Sozialwissenschaftlers, dem zwar die Kompetenz zugestanden wird, ein Idealkonzept entwerfen zu können, aber die Kompetenz abgesprochen wird, zur Umsetzung dieses Konzepts in der Praxis beitragen zu können[40]. Trotz dieses relativen Charakters behaupteter absoluter Handlungsgrenzen, kann nicht übersehen werden, daß es tatsächlich absolute Handlungsgrenzen für die

[40] Dies endet oftmals in einer beidseitigen Kränkung: Die Sozialwissenschaftler fühlen sich gekränkt, weil über ihre theoretischen Verfeinerungen und Bemühungen mit einer Nonchalance hinweggegangen wird, die an ihre berufliche Ehre geht. Die Praktiker dagegen sehen das von den Sozialwissenschaftlern vorgelegte Idealkonzept oftmals als Pauschalkritik an ihrer Arbeit an, einer Kritik, die ihre alltäglichen Mühen und Bemühungen nicht achtet.

berufliche Ausbildung gibt, deren Überwindung letztendlich auf ihre Selbstüberwindung hinauslaufen würde. Diese absoluten Grenzen liegen zuvorderst in der vermittelbaren Stoffmenge und deren Reflexions- und Abstraktionsniveau.

Beide Grenzen werden sowohl durch die Möglichkeiten der institutionellen Ausbildungsorganisation als auch durch die legitimen Ausbildungsansprüche der Lernenden markiert. Im folgenden werden diese Grenzen und die mit ihnen einhergehenden Transformationen bezüglich der von uns vorgeschlagenen neuen Fachlichkeit und Beruflichkeit diskutiert. Der Diskussion liegen Aussagen von Praktikern zugrunde, mit denen wir ständig konfrontiert wurden. Wir haben sie zu idealtypischen Aussagen verdichtet, auf die wir unsere Position dialogisch beziehen:

"Wir haben das Problem, daß sich unsere Fachlichkeit ständig wandelt, neue Programmiersprachen und Rechnerarchitekturen verlangen eine ständige Veränderung der Fachlichkeit. Dabei ist die Stoffmenge, die hinzugefügt wird, meist größer als die, die ersetzt wird. Der Stundenplan ist randvoll, und jetzt kommen Sie mit einer nochmals riesigen Stoffmenge. Die können wir doch gar nicht mehr integrieren, höchstens auf Kosten harten technischen Wissens und Könnens, und das erscheint uns kaum verantwortbar."

Die Fachlichkeit von IT-Fachkräften ist tatsächlich in einem besonderen Maße von der Dynamik, die daraus entsteht, ständig neues technisches Wissen integrieren zu müssen, geprägt. Diese Dynamik ist auch nicht durch die Orientierung der Fachlichkeit an dem Qualifikationsprofil Ganzheitliche Arbeitsgestaltungskompetenz aufzufangen. Der Anpassungszwang an einen höchst dynamischen »State of the Art« ist gewissermaßen das Schicksal jeder informationstechnischen Ausbildung.

Offen bleibt allerdings, wie diese Anpassungsleistungen erfolgen. Dahinter steht die Frage, ob die Fachlichkeit zum beliebigen Spielball informationstechnischer Neuerungen wird, oder ob sie diese Neuerungen auf der Folie einer kohärenten Fachlichkeit interpretiert und integriert. Die erste Option führt dazu, daß die Fachlichkeit zunehmend diffundiert, tendenziell zu einem Flickenteppich wird, dessen einziger Zusammenhang eine angenommene Verwertbarkeit auf dem Arbeitsmarkt darstellt. Da die Dynamik informationstechnischer Entwicklungen nicht auf die Ausbildungszeit beschränkt bleibt, wird dem Auszubildenden auf seinen beruflichen Weg noch die Pflicht mitgegeben, sich ständig weiterzubilden. Ist die Ausbildung wenigstens noch halbwegs kohärent und ihre Tendenz, einem Flickenteppich zu gleichen, durch die Notwendigkeit mathematischen und informationstechnischen Grundwissens noch begrenzt, so ist die Weiterbildung

diesbezüglich doppelt frei: Ihre Kurse sind auf kein Konzept bezogen, und in ihren Kursen wird keine konzeptionelle Orientierung geboten.

Mit dem Vorschlag, die Fachlichkeit an dem Qualifikationsprofil Ganzheitliche Arbeitsgestaltung auszurichten, verfolgen wir die Ziele,

1. neue informationstechnische Inhalte auf den Arbeitsprozeß zu beziehen, das heißt, sie in ihrer Bedeutung bewertbar zu machen und somit umzuinterpretieren,
2. den Auszubildenden weiterbildungsfähig zu machen, indem er neue informationstechnische Entwicklungen in ihrer Relevanz für den Prozeß der Software-Entwicklung einzuschätzen lernt, und sie situationsangemessen anpaßt und anwendet.

Der Anspruch der Ganzheitlichen Arbeitsgestaltungskompetenz als übergreifendes Leitbild der Fachlichkeit bedeutet, daß die bestehende Fachlichkeit uminterpretiert und auf die Sichtweise des Software-Entwicklungsprozesses als sozialen Prozeß bezogen wird. Wie dies ansatzweise geschehen kann, haben wir in dem Qualifizierungsbaustein »Aufgabenverständnis« und dem Rahmencurriculum »Moderne Software - Entwicklung« entwickelt, die allerdings nur einen Einstieg in eine umfassende Uminterpretation der Fach- und Beruflichkeit darstellen können.

Diese umfassende Uminterpretation traditioneller Fachlichkeit, die wir als Praxis einer »reflexiven Verwissenschaftlichung« bezeichnen, da sie auf einer Reflexion technisch-wissenschaftlicher Erkenntnisse, Methoden und Artefakte bezüglich ihrer sozialen Gestaltung beruht, führt zu einer neuen Gewichtung einzelner Inhalte bisheriger Fachlichkeit. Eine Überfrachtung der Stoffmenge durch die von uns vorgeschlagene Neuausrichtung der Fachlichkeit ist keineswegs zwingend. Vielmehr liefert das neue Paradigma im Zusammenhang mit Trendanalysen auch Kriterien, die der Entrümpelung des Curriculums dienen.

"Die Uminterpretation traditioneller Fachinhalte ist doch oftmals gar nicht möglich. Zum Beispiel: Eine Programmiersprache ist eine Programmiersprache ist eine Programmiersprache ... . Sie bleibt es doch immer und folgt ihrer eigenen Logik. Wie wollen Sie diese Logik durchbrechen und den sozialen Prozeß der Software-Entwicklung ins Spiel bringen?"

Das ist ein heikler, aber auch entscheidender Punkt. Wir wollen dieser Frage im folgenden dadurch gerecht werden, daß wir die Bedeutung von Programmiersprachen für die Informatik theoretisch klären, auf Bedeutungsverschiebungen, die sich durch einen veränderten Nutzungskontext und Tendenzen bei der Ent-

wicklung neuer Programmiersprachen ergeben, hinweisen und entsprechende didaktische Konsequenzen ziehen.

Eine Programmiersprache ist eine Programmiersprache ist eine Programmiersprache ... . Daran kommt man nicht vorbei. Aber das Entscheidende ist, daß Programmieren zunehmend nicht mehr als Schreiben eines Programmcodes mit einer Programmiersprache begriffen werden kann. Nach den Worten des in der Informatik tätigen Logikers Siefkes ist selbst der Begriff der Programmiersprache schon irreführend:

"Wenn Computer nicht reden können, sollte man Programmierkalküle nicht »Sprachen« nennen. Solche Formalismen sind wie die mathematischen und logischen der natürlichen Sprache nachgebildet, aber per definitionem, nicht durch Entwicklung. Sie ändern sich nicht, wir müssen sie ändern. Richtig gebraucht sind sie Werkzeuge, Kategorie »Hilfsmittel«, nicht »Sprache«. Mißbraucht werden sie zu Fallen." [Siefkes 1993, S. 129].

Die wissenschaftliche Vorstellung strukturalistischer Linguistik, daß sich die natürlich-sprachliche Beschreibung von Realität durch Methoden formalisieren oder gar generieren lassen würde, ist zunehmend fragwürdiger geworden. "The new techniques of structural analysis earned for linguistics the reputation of the most of the social-science disciplines. But these techniques proved to be quite limited in their application. (...) Minor dialect variations, loan words, and other diffusionary phenomena were also frequently desmissed as either marginal or not relevant for formalisation." [Gumperz/ Hymes 1986, S. 8]. Sprache hat als sprachliche Interaktion, in der maßgeblich Realität konstruiert wird[41], Prozeßgestalt. Als Prozeß entzieht sich Sprache der Formalisierung mittels formalisierter Sprachen. Programmiersprachen dienen allein dazu, formalisiertes Wissen in formalisiertes Wissen, das elektrotechnisch bearbeitbar ist, zu transformieren. Die Bemühungen um eine Annäherung neuentwickelter Programmiersprachen an die »natürliche Sprache«[42] mögen diesen Prozeß der Transformation von

---

[41] "Finally, I explore the ways that language can be used as an active, strategic tool in social interaction. Through this perspective it becomes possible to see how meaning is created in interaction, rather than being simply a reflex of given situation." [Beeman 1982, S. xii].

[42] Es handelt sich selbstverständlich nur um das strukturalistische Bild der natürlichen Sprache und nicht um lebendige Sprache. Ausgeblendet wird damit das, was Hymes in seiner Schrift "Perspektiven für eine (Ethno-) Linguistische Theorie" für eine umfassende Theorie einfordert, indem er programmatisch verlangt, "daß jede konsequente linguistische Theorie notwendigerweise eine Theorie ist, die ethnographische Voraussetzungen macht. Das heißt, jede linguistische Theorie, die diese Bezeichnung verdient, formuliert notwendigerweise nicht nur Annahmen über die Sprachsysteme, sondern auch über ethnolinguistische Systeme, über die partielle Abhängig-

Formalem erleichtern, lösen aber nicht das Problem der »ursprünglichen« Formalisierung.

Dieses Dilemma der Informatik, daß Formalisierung notwendig, aber formal nicht bewältigbar ist, spielte für sie solange keine Rolle, als sich die praktische Informatik meist nur auf formalisiertes Wissen bezog, das heißt, das Problem der Formalisierung anderen überließ. Da es moderne Software-Entwicklung in zunehmenden Maße nicht mehr mit der Abbildung vorab strukturierter Arbeits- und Produktionsabläufe zu tun hat, welche in Form von formalisiertem Wissen durch Betriebsingenieure und Verfahrenstechnikern aufgearbeitet worden waren, wird die »ursprüngliche« Formalisierung verstärkt zu ihrem zentralen Problem. Diese aus dem veränderten Nutzungskontext von IT-Systemen resultierende Tendenz korrespondiert mit dem Trend zu sogenannten »4GLs«, die zumindest den Anspruch haben, leichter erlern- und benutzbar zu sein. Ebenso findet sich die Abkehr von der »Kernqualifikation« der Beherrschung von Programmiersprachen auch durch wirtschaftswissenschaftliche Untersuchungen bestätigt: "Der relativ geringe Einfluß der Sprachkenntnisse auf die Produktivität ist eine wichtige Erkenntnis, die durchaus nicht intuitiv ist. Den Stellenanzeigen für Programmierer nach zu urteilen, werden Sprachkenntnisse und auch Anwendungserfahrung von den meisten EDV-Leitern überbewertet"[Fisher 1990, S. 6].[43]

Welche didaktischen Folgerungen für die Rolle der Programmiersprachen in einer neubestimmten Fachlichkeit für IT-Fachkräfte können aus diesen Überlegungen abgeleitet werden? Reicht es aus, den Auszubildenden die avanciertesten Reflexionen des eigenen Faches zur Verfügung zu stellen, so z.B. die programmatische Schrift Nygaards "Program Development as a Social Activity" mit der zentralen Aussage: "to program is to understand!", den hermeneutischen Ansatz von Winograd und Flores, die Sichtweise Naurs - "Programming as Theory Building" - oder Floyds "Software Development as Reality Construction"? Oder sollte der Umfang der Programmiersprachen innerhalb des Lehrplans generell erheblich gekürzt werden?

Die didaktischen Entscheidungen, die zur Beantwortung dieses Fragenkomplexes, der auf eine Uminterpretation traditioneller Fachlichkeit zielt, getroffen werden müssen, sind u.E. nur aufgrund einer systematisierten Problemsicht möglich. Bezüglich dieser »formalisierten« Problemsicht können dann verschiedene

---

keit zwischen Eigenschaften (wie auch immer eng definierter) sprachlicher Systeme einerseits und den Charakteristika derer, die sie anwenden, sowie den Bedingungen ihres Gebrauchs andererseits."[Hymes 1979 S. 99].

[43] Fisher bezieht sich auf eine empirische Studie von Boehm und Papaccio [vgl. Boehm 1988, S. 1466].

Präferenzen formuliert werden, die bestimmte Formen der Beruflichkeit implizieren. Ein solches systematisches Schema stellt z.B. folgende Tabelle dar.

Systematische Einbettung der "Programmiersprachen" in eine umfassende Fachlichkeit »Formalisierung«

	Wissensformen		
	Erschließungswissen[44]	**Anwendungswissen**	**Reflexionswissen**
Inhalt	Formale Logik und Mathematik[45] (Grundlagen der Formalisierung und Transformierung von Formalismen)	konkrete Programmiersprache und Probleme ihrer konkreten technischen Anwendung	Grenzen und Möglichkeiten des Formalisieren (sozialer Prozeß und technisches System)
berufliches und personales Bildungsziel	Weiterbildungsfähigkeit (Problemlösefähigkeit)	Routine (Können, Beherrschen)	situationsadäquates Handeln Persönlichkeitsbildung
akzentuierte Präferenz für die Beruflichkeit	Professionalisierung	Arbeitsteilige Spezialisierung	Verwissenschaftlichung

Unser Vorschlag einer Uminterpretation traditioneller Fachlichkeit heißt im Falle der Programmiersprachen, daß das hier genannte Erschließungs- und Reflexi-

---

[44] Exakter wäre: Wissen, das zur Erschließung absolut notwendig ist. (Neben den mathematischen und logischen Grundlagen wäre hier auch hardwaretechnisches Grundwissen bis hin zum Verstehen von Betriebssystemen zu nennen. Da dieses Wissen aber nicht von gleicher fundamentaler Qualität ist und einer anderen Verzeitlichung folgt, wollen wir es hier nicht aufnehmen: Es steht gewissermaßen zwischen dem Erschließungs- und Anwendungswissen).

[45] Es gibt in der Informatik eine seit Jahren mit hohem emotionalen Einsatz geführte Diskussion über die Bedeutung, die formale Logik und Mathematik für das Fach haben. Ausgelöst wurde diese Debatte durch den Aufsatz "On the Cruelity of Really Teaching Computing Science" von Dijkstra [Dijkstra 1989], in dem er eine harte (grausame) mathematische Grundbildung als einzig adäquaten Lehrweg beschreibt und jede andere Perspektive als »Gefälligkeitsproblem« und somit für die Grundbildung als redundant verwarf. Die Antworten derjenigen, die die Gestaltung von Computerprogrammen als die Essenz des Faches begreifen, fielen entsprechend hart aus [vgl. zu dieser Debatte, Coy, 1992, S. 1ff.]. Interessant für uns ist, daß sich beide Positionen in ihrer Kritik an der Bedeutung von Programmierkursen einig waren, Dijkstra aber gewissermaßen eine »Erschließungsperspektive« und seine Kritiker eine »Reflexionsperspektive« einnehmen. Eine Verständigung ist somit kaum möglich.

onswissen zum integralen Bestandteil der Ausbildung werden muß. Ohne diese Einbettung ist ein Verständnis von Software-Entwicklung als sozialem Prozeß unmöglich, da der Tätigkeit des eigentlichen Programmierens jeglicher Bezug zu diesem Prozeß fehlt.

"Je konkreter die inhaltliche Ausgestaltung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« formuliert wird, um so deutlicher wird auch, daß dieses Qualifizierungsziel nicht nur neue, interdisziplinäre Wissensgebiete aufschließt, sondern auch extrem hohe Anforderungen an das Reflexionsvermögen und Theorieverständnis der Auszubildenden stellt. Wer sich aber gerade nicht für eine universitäre Ausbildung entscheidet, dokumentiert damit doch genau dieses, daß er sich nicht auf dieses Abstraktionsniveau begeben möchte. Des weiteren verfolgen Sie mit ihrem Konzept der Ganzheitlichen Arbeitsgestaltungskompetenz auch das Ziel, die Weiterbildung von IT-Fachkräften zu strukturieren und inhaltlich auszugestalten. Damit stellt sich für uns die Frage, wie eine weniger theoretisch, sondern mehr praktisch ausgerichtete Aus- und Weiterbildung das Qualifizierungsziel »Ganzheitliche Arbeitsgestaltungskompetenz« umsetzen kann."

Die Frage nach den Bildungsansprüchen der Lernenden ist für uns ein doppeltes Problem: Einerseits kommen die Menschen mit vorberuflich sozialisierten Ansprüchen in bestimmte Aus- und Weiterbildungsinstitutionen, mit denen sie mehr oder weniger konkrete Bildungsangebote für sich verbinden, andererseits verlassen sie diese Einrichtungen mit anderen Ansprüchen und Bildungsinteressen. In den Aus- und Weiterbildungsinstitutionen verändern sich die Bildungsansprüche der Lernenden, wobei die Institutionen dabei eine oft sehr zweischneidige Rolle spielen: Die von ihnen geweckten Bildungsansprüche sollen nur solche sein, die sie selber abdecken können. Der Ausgebildete soll glauben, daß er in der Ausbildung alles gelernt hat, was er braucht. Der Interessent an Weiterbildung wird ebenfalls mit einem sehr konkreten Verwertungsversprechen gelockt. Dieses Verfahren führt immer zu einem Ausschluß, zu einem Mißtrauen gegenüber allgemeiner Bildung, universitärer Bildung usw., einem Ausschluß, der nicht selten mit den Ansprüchen der Lernenden selbst legitimiert wird. Aber es werden nicht nur die andere, schon immer als Konkurrenz empfundene institutionalisierte Bildung ausgeschlossen, sondern, was in unserem Fall von größerer Bedeutung ist, auch die Möglichkeiten und Chancen selbstorganisierter Lernprozesse.

Die Umsetzung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« in der beruflichen Aus- und Weiterbildung beruht dagegen maßgeblich darauf, daß das Fundament für selbstorganisierte Lernprozesse auf Seiten des Individuums gelegt wird. Die dafür von uns entwickelten Ansatzpunkte sind Ge-

genstand des folgenden Kapitels. Sie unterscheiden sich deutlich vom Konzept der Schlüsselqualifikationen, deren Verdienst allerdings, das Bildungsziel »Lernen des Lernens«[46] in die berufliche Ausbildung hineingetragen zu haben, hier nicht geschmälert werden soll.

Diese grundsätzlichen Überlegungen zum institutionalisierten Umgang mit Bildungsansprüchen entheben uns natürlich nicht der Beantwortung der Frage, inwieweit das Qualifizierungsziel »Ganzheitliche Arbeitsgestaltungskompetenz« eine Intellektualisierung der Lernwege impliziert. Als erstes stellen wir fest: Die mit dieser Uminterpretation der traditionellen Fachlichkeit einhergehende Verwissenschaftlichung und Professionalisierung führt in der Tat zu einer Intellektualisierung der Ausbildung. Wenn man aber anerkennt, daß diese Verwissenschaftlichung und Professionalisierung das Schicksal einer verantwortungsbewußten Ausbildung ist, die eine »Proletarisierung« [vgl. Hartmann 1993, S. 392ff.] und Marginalisierung der Ausgebildeten im Arbeitsprozeß vermeiden will, muß man sich dem Problem der Intellektualisierung stellen. Inwieweit allerdings Verwissenschaftlichung und Professionalisierung traditioneller Intellektualisierung (Begriffs- und Theoriebildung) verpflichtet sind, ist eine offene Frage. Immerhin ist es nicht nur denkbar, sondern es gibt auch Ansätze, "über die Erfahrung zum Begriff" [Bojanowski 1991, S. 119] zu kommen oder in der praktischen Arbeit einen experimentellen Zugang zur Wirklichkeit[47] zu gewinnen. Neben der didaktischen Grundlegung selbstorganisierter Lernprozesse ist eben Genanntes Thema des folgenden Kapitels.

2.4 Das »Dreieck beruflicher Handlungskompetenz« als didaktischer Ansatzpunkt zur Umsetzung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz«

Der Begriff der beruflichen Handlungskompetenz und seine Folgen für die Kritik am behavioristischen Lernmodell

Darüber, daß das Ziel der beruflichen Ausbildung im Erwerben der beruflichen Handlungskompetenz liegt, läßt sich schnell ein Konsens erreichen. Die berufliche Handlungskompetenz an sich ist in einem gewissen Sinn ein tautologischer Begriff, der nichts anderes besagt, als daß der Arbeitende das können muß, was er können muß. Mit dem theoretischen Hintergrund der Handlungsregulationstheorie gewinnt der Begriff der beruflichen Handlungskompetenz allerdings

---

[46] Steht aber auch wörtlich schon so bei W.v.Humboldt [vgl. Heydorn 1980, S. 260].

[47] Ein weiterer didaktischer Grund, dem »Prototyping« eine herausragende Stelle im Curriculum Software Engineering einzuräumen!

eine analytische Dimension, die nachhaltige Folgen für die Entwicklung einer ihr angemessenen Didaktik hat.

Der Kerngedanke der Handlungsregulationstheorie besteht in der Überwindung der statischen Vorstellung, daß berufliches Handeln nichts weiter als die einfache Anwendung von Wissen und Fertigkeiten darstellen würde. Vielmehr, so lautet die zentrale These, ist Handeln nur als Prozeß versteh- und untersuchbar. Die Prozeßhaftigkeit einzelner Handlungen ist allein schon dadurch bedingt, daß Handlungseinheiten immer in einem größeren Handlungsgefüge stehen. Dieses größere Handlungsgefüge ist aber weder durch reines Reagieren auf eine objektive Situation, noch durch reines Denken, das heißt, durch ein abrufbares allumfassendes Handlungsschema bestimmt, sondern dieses Handlungsgefüge ändert sich permanent im Prozeß der Planung einer Handlungsstruktur und ihrer Kontrolle mittels der praktischen Durchführung einzelner Handlungseinheiten.

Die Notwendigkeit, Handlungen ständig zu regulieren, liegt schon allein darin begründet, daß das Wissen und die Fertigkeiten, über die ein Individuum verfügt, immer einen allgemeinen Charakter haben, während konkrete Handlungen immer einzigartig sind. Dieses permanente Regulationserfordernis, das »Aussuchen« passender verinnerlichter Handlungsschemata[48], das Modifizieren und Verbinden einzelner Handlungseinheiten, wird nun keineswegs allein situativ bestimmt, sondern steht in einem übergreifenden Zusammenhang, dem Handlungsgefüge. Die Handlungsregulationstheorie liefert die Perspektive und die Instrumente zur funktionalen Bestimmung einzelner Handlungseinheiten im Handlungsgefüge. So werden Handlungseinheiten beispielsweise bestimmten, in einem hierarchischen Verhältnis stehenden Ebenen im Handlungsgefüge zugeordnet oder bezüglich ihrer sequentiellen Abfolge bestimmt. Gleichzeitig berücksichtigt die Handlungsregulationstheorie einen weiteren Aspekt des Handelns, der oftmals nur losgelöst vom Wissens- und Fertigkeitenaspekt betrachtet wird: "Dieses Handlungsregulationsmodell ist nicht nur ein kognitives Modell, es bezieht sich durch die Rolle der Ziele als geistige Vorwegnahmen und Vornahmen zugleich auch auf motivationale Aspekte des Handelns" [Höpfner 1991, S. 18].

---

[48] Insofern stellt Volpert, ein Hauptvertreter der Handlungsregulation fest, "daß die Handlungsregulationstheorie ein »Speichermodell« braucht, aus welchem sich ihr Prozeßmodell stringent ableiten läßt" [Volpert 1983, S. 8]. Wenn wir den innerpsychischen Speicheraspekt von Handlungseinheiten betonen wollen, sprechen wir im folgenden von Handlungsschemata, wobei davon auszugehen ist, daß nur wenige basale Handlungseinheiten als gespeicherte Handlungsschemata vorliegen, und der größte Teil von Handlungseinheiten aus abstrakteren Handlungsschemata **konstruiert** wird [vgl. ebd., S. 9ff.].

Ein aus der Theorie der Handlungsregulation abgeleitetes didaktisches Modell kann u.E. zum Ausgangspunkt der Entwicklung von Ansatzpunkten zur Umsetzung des Qualifizierungsziels »Ganzheitlicher Arbeitsgestaltungskompetenz« werden.[49] Die im nachfolgenden Kapitel unternommene Analyse dient der theoretischen Klärung dieser Ansatzpunkte. Diese Klärung erscheint uns schon deshalb notwendig, weil mit der von uns vorgeschlagenen Neubestimmung der Fachlichkeit und Beruflichkeit auch der pädagogische Handlungsbereich aus einer neuen Perspektive gesehen wird. Das heißt: Eine Umorientierung in der Fachlich- und Beruflichkeit erfordert nicht nur ein neues didaktisches Konzept, sondern impliziert auch eine Kritik des traditionellen pädagogischen Handlungsbereichs der Ausbilder.

Im Zentrum des pädagogischen Handlungsbereichs steht der Unterricht, beziehungsweise die Unterweisung. Ziel des pädagogischen Handelns ist aber nicht der Unterricht, sondern die intentionale Veränderung des lernenden Individuums. Beide Aussagen für sich genommen sind banal. In ihrem Zusammenspiel verweisen sie dagegen auf das Problematische des pädagogischen Handelns: Das lernende Subjekt selbst entzieht sich dem pädagogischen Handlungsbereich, wird nur über die Gestaltung des Mediums Unterricht erreicht. Die traditionelle behavioristische Sicht des pädagogischen Handlungsbereichs[50], die selbst in ihrer modifizierten Form des Modellernens die Differenz von Lehren und Lernen verwischt, mag genügen, um sogenannte rezeptive und mechanistische

---

[49] Söltenfuß geht sogar so weit zu behaupten, daß über einen handlungspsychologischen Ansatz der Persönlichkeitsentwicklung "zum ersten Mal ein weitgehend theoretisch geschlossener konsistenter Rahmen für eine Didaktik beruflicher Lernprozesse geliefert" worden sei [Söltenfuß zit.n. Ebner 1988, S. 387]. Auch wenn Söltenfuß sein theoretisches Konzept nicht allein auf die Grundannahmen der Handlungsregulationstheorie bezieht, so ist diese Theorie für ihn doch der Schlüssel zur Entwicklung einer konsistenten Didaktik (zur Kritik der verkürzten Rezeption der materialistischen Psychologie durch Söltenfuß vgl. [Ebner 1988, 387ff]).

[50] Adam bezeichnet diese Sichtweise als technologische didaktische Positionen. Die traditionelle lerntheoretische Didaktik geht nach dem Schema Intentionalität - Thematik - Methodik - Medienwahl vor. Dieses Schema wird auf den Wissensstand und die vermutete Lernfähigkeit der Lernenden als Bedingungsfeld bezogen und nach drei Prinzipien bewertend strukturiert: dem Prinzip der Interdependenz (der widerspruchsfreien Wechselwirkung aller Planungsmomente), dem Prinzip der Variabilität (das methodische Vorgehen muß alternatives Schülerverhalten zulassen) und dem Prinzip der Kontrollierbarkeit (Überprüfung des ursprünglich intendierten Lernziels). Diese lerntheoretische Didaktik hat aus zwei Richtungen Kritik erfahren. Einerseits versucht die kybernetische und systemtheoretische Didaktik, die lerntheoretische Didaktik zu objektivieren und technologisch verfügbar zu machen, andererseits kritisieren Vertreter einer an der Kritischen Theorie angelehnten subjektorientierten Didaktik den instrumentell-technologischen Charakter lerntheoretisch fundierter Didaktiken. Beide Kritiken stehen sich diametral gegenüber [vgl. Adam 1988].

Lernprozesse in Gang zu setzen. Entdeckendes und sinnbezogenes Lernen, das auf ein reflektierendes Verstehen abzielt, kann so kaum absichtsvoll initiiert werden.[51] Die explizite oder implizite Orientierung am behavioristischen Lernmodell versperrt letztendlich die Umsetzung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz«. Das behavioristische Paradigma des programmierten Unterrichts, dessen übergeordnetes Ziel es ist, den Lernenden in vorbestimmten kleinen Lernschritten auf ein definiertes Verhaltensziel hinzuführen, zeigt sich in der beruflichen Bildung im klassischen methodischen Prinzip des Lehrgangs. "Der *Lehrgang* als systematisch aufgebauter synthetischer Gang des Lernens, dessen innere Logik nicht der Sache, sondern z.B. dem Prinzip »vom Einfachen zum Komplizierten« und »vom Teil zum Ganzen« folgt. Der Lernende kann dabei erst im Rückblick die Teile zu einem sinnvollen Ganzen ordnen" [Bojanowski 1991, S. 87]. Diese behavioristische Sichtweise des Lernens und Lehrens führt auch zur Vermittlung einer bestimmten Sichtweise der Fachlichkeit, die zwar nicht intentional vermittelt, aber als unbeabsichtigtes Nebenprodukt fast zwangsläufig sozialisiert wird. Für die Informatikausbildung an den Universitäten beschreibt Wendt das »behavioristische« Prinzip und die Folgen eines solchen Vorgehens:

"Bereits in der Schule hat der spätere Informatikstudent seine ersten Programme gebastelt; in den ersten Wochen seines Studiums läßt man ihn dann einfache Programme »konstruieren«, und im Laufe des Studiums werden seine Programme immer komplexer, so daß er dann in dem Glauben abgehen kann, er habe die höchste Stufe der Komplexitätsbeherrschung erreicht. Wollte man die Ingenieurausbildung in entsprechender Weise gestalten, dann müßte man dem Elektrotechnik-Studenten, der als Schüler bereits eine Steuerung für seine elektrische Eisenbahn gebastelt hat, in den ersten Wochen seines Studiums einen kleinen Verstärker konstruieren lassen, und müßte dann im Laufe seines Studiums immer komplexere Apparate konstruieren."[Wendt 1993, S. 38]. Die unbeabsichtigten Folgen davon: "Dem Informatikstudenten wird (...) suggeriert, der Übergang vom »Programmieren im Kleinen« zum »Programmieren im Großen« könne im wesentlichen durch die Hinzunahme einiger angemessener programmiersprachlicher Konstrukte bewältigt werden. Niemand weist ihn darauf hin, daß die schwierigsten Probleme des »Großen« gar keine »Programmierer«-Probleme sind." [ebd.].

---

[51] Die Unterscheidung bei Typen von Lernprozessen durch die zwei Dimensionen rezeptives vs. entdeckendes Lernen und sinnbezogenes vs. mechanisches Lernen geht auf Ausubel zurück [vgl. Ausubel 1973].

Die Orientierung des didaktischen Vorgehens an dem Begriff der beruflichen Handlungskompetenz und seiner theoretischen Fundierung durch die Handlungsregulationstheorie und deren kognitionspsychologischen Implikationen verspricht dagegen, gleichermaßen der Logik der Sache **und** dem lernenden Subjekt gerecht zu werden. Die Logik der Sache, das heißt, die Ganzheitliche Arbeitsgestaltungskompetenz, verlangt ein didaktisches Umdenken, das vom lernenden Subjekt ausgehen muß. Daß mit dem Begriff der beruflichen Handlungskompetenz freilich kein fertiges didaktisches Konzept vorliegt, das nur noch umgesetzt werden müßte, sondern vorerst »nur« analytische Begriffe verbunden sind, die eine angemessene Grundlage des didaktischen Vorgehens bilden können, mag den Praktiker zunächst skeptisch stimmen.[52] Wir hoffen jedoch, zeigen zu können, daß sich daraus praktische pädagogisch-didaktische Handlungsorientierungen ableiten lassen.

2.4.1 Berufliche Handlungskompetenz als analytischer Begriff

Der Begriff der beruflichen Handlungskompetenz kann grundsätzlich unter zwei »analytischen« Gesichtspunkten betrachtet werden:

1. Berufliche Handlungskompetenz gilt als das Ganze, das analysiert werden soll. Das analytische Erkenntnisinteresse versucht, **einzelne Dimensionen der beruflichen Handlungskompetenz**, wie verschiedene Formen des Wissens, Fertigkeiten, soziale Kompetenzen, Motivationen und Einstellungen, systematisch zu unterscheiden, ihren Zusammenhang untereinander zu klären und die Bedeutung einzelner Dimensionen für das Ganze (Entwurf einer Handlungsstruktur und ihre Durchführung) **funktional** zu bestimmen.[53]
2. Berufliche Handlungskompetenz wird als die **personale Dimension im historisch-gesellschaftlich bestimmten Arbeitsprozeß**[54] angesehen, und

---

[52] Die Bemühungen der Theorie stoßen auf die skeptische Ungeduld der Praxis. Diese Grunderfahrung eines praktisch orientierten Forschungsprojektes ist fundamental. Sie kann u.E. auch nicht durch sprachliche Angleichungen, die die unterschiedliche Realitätsperspektive verschleiern, bewältigt werden. Der einzige adäquate Umgang mit diesem Problem besteht in einer durch gegenseitige Achtung gekennzeichneten Kommunikation. So gesehen sind das Ziel und die Struktur eines anwendungsorientierten, kooperativen Forschungsprojekts durchaus mit einem Softwareprojekt vergleichbar: Es geht um eine Neugestaltung von Arbeit, die nur in einem gemeinsamen Kooperationsprozeß unterschiedlicher Experten bewältigt werden kann.

[53] Auf dieser Ebene bewegen sich die traditionellen Vertreter der Handlungsregulationstheorie.

[54] Diese Perspektive gerät bei den traditionellen Vertretern der Handlungsregulationstheorie gänzlich in den Hintergrund. So kritisiert beispielsweise Simon die Handlungsregulationstheorie Hackers: "Arbeit wird auf der Ebene der Erscheinungsformen

sie wird in Beziehung gesetzt zu Begriffen wie Handlungsspielraum**,** Handlungsraumkonzept, berufliches Handlungsvermögen, berufliche Handlungssicherheit [vgl. Heeg 1988, S 19f].

Diese zwei »analytischen« Gesichtspunkte können durch einen dritten, in dem sie gewissermaßen **synthetisiert** sind, ergänzt werden:

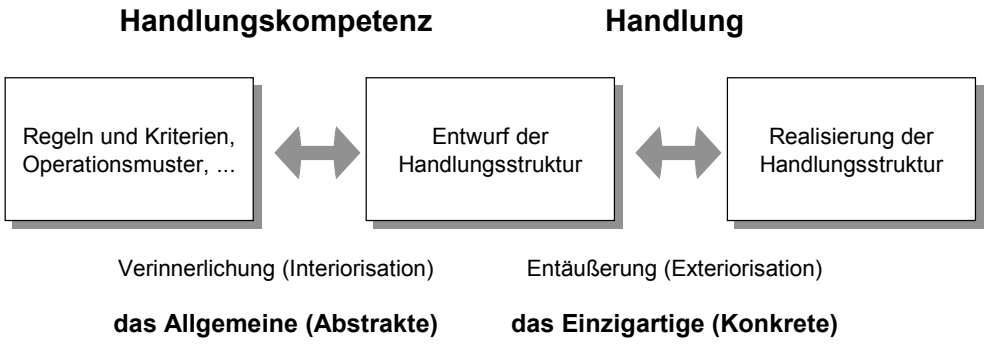
3. Die Bedeutungen einzelner Dimensionen der beruflichen Handlungskompetenz für den Handlungsspielraum, für das Handlungsraumkonzept, für das berufliche Handlungsvermögen und den Arbeitsprozeß insgesamt, werden zum Ausgangspunkt eines komplexen, dialektischen Verständnisses von beruflicher Handlungskompetenz. Dieser dritte Aspekt ist deshalb von großer Bedeutung, weil hier über **die Wichtigkeit einzelner Dimensionen der beruflichen Handlungskompetenz für die Bewältigung der Aufgaben des historisch-gesellschaftlichen Arbeitsprozesses** diskutiert wird.

Bevor wir die berufliche Handlungskompetenz von IT-Fachkräften unter diesen drei Gesichtspunkten betrachten, um didaktische Strategien zur Umsetzung unseres Qualifizierungsziels zu finden, soll allerdings das Verhältnis von beruflicher Handlungskompetenz und Handlung auf der »tautologischen« Ebene reflektiert werden. Das heißt, es geht uns vorerst darum, das Entsprechungsverhältnis von inneren Handlungsplänen und äußeren situationsspezifischen realen Handlungen nochmals zu problematisieren. Die Sensibilisierung für diese Problematik bildet u.E. gewissermaßen ein »theoretisches Schlüsselerlebnis« zum Verständnis der Notwendigkeit einer didaktischen Umorientierung. Das nachfolgende Modell zeigt die Strukturanalogie von Handlungskompetenz, Entwurf der Handlungsstruktur und eigentlicher Handlung und verweist auf ihre prozeßhaften Verbindungen in Form der Interiorisation (Verinnerlichung) und Exteriorisation (Entäußerung).

---

als industrielle Tätigkeit in einem »Mensch-Maschine-Funktionssystem« beschrieben, ohne Reflexion auf ihren konkret historischen, gesellschaftlichen Konstitutionszusammenhang" [Simon 1981, S. 508]. Wir folgen in diesem Punkt zwar der Kritik Simons, gehen aber mit Schurer davon aus, daß diese Kritik nicht zu einer generellen Verwerfung der Handlungsregulationstheorie führen muß, da die historisch-gesellschaftliche Arbeitsprozeßperspektive (ebenso wie die kognitionspsychologische Perspektive) ergänzend integriert werden kann [vgl. Schurer 1986, S. 684].

Zusammenhang von Handlungskompetenz, Handlungsstruktur und Handlung



Quelle: nach Höpfner 1991, S.21; erweitert um die Dimension »Allgemeines« und »Einzigartiges«.

Die Begriffe Interiorisation und Exteriorisation bezeichnen die gewissermaßen »innerpsychischen« Lokalitäten des Handlungsprozesses, der, da er ständig innere und äußere Handlungen rückkoppelt, immer auch ein Lernprozeß ist. Da eine ausführliche Besprechung der kognitionspsychologischen Grundlagen dieses Lernprozesses den Rahmen unserer Arbeit sprengen würde, erscheint es uns angebracht, den Blick auf das für uns Wesentliche des Modells zu richten, der naheliegenden **Differenz zwischen dem Allgemeinen der verinnerlichten Handlungspläne und dem Einzigartigen der ausgeführten Handlung**. Daß dieses, von uns als Wesentliches ausgemachte, den meisten Vertretern des handlungstheoretischen Modells selbst oftmals verborgen bleibt, beziehungsweise ihnen nicht der Rede Wert zu sein scheint, liegt nicht zuletzt an deren pragmatistischer erkenntnistheoretischer Orientierung: Wenn die Handlungskompetenz den inneren Entwurf einer Handlungsstruktur ermöglicht, deren Realisierung zum gewünschten Ereignis führt, liegt erfolgreiches Handeln vor. Die Frage nach der wesensmäßigen Differenz zwischen den erfolgreichen Handlungsplänen und der in einer einzigartigen Situation einzigartig ausgeführten Handlung erscheint, da keine Regulationserfordernisse vorliegen, als philosophische Spitzfindigkeit. Im Falle des Arbeitsprozesses Software-Entwicklung kommt dieser Differenz von Allgemeinem und Einzigartigem allerdings eine besondere praktische Relevanz zu. Allgemeinen individuellen Handlungsplänen stehen einzigartige Situationen gegenüber, die eine einfache Anwendung unmöglich machen:

- Die Einzigartigkeit eines Softwareprojekts ist allein schon durch die **Möglichkeit der beliebigen Duplizierbarkeit** von Software gegeben. Es wird nur neue Software entwickelt. Dies gilt im strengen Sinne genauso für die Wartung von Software.
- Die realisierten Handlungen im Softwareprojekt kommen nicht aufgrund von individuellen Handlungsplänen zustande, sondern durch **das Aufeinanderbeziehen unterschiedlicher Handlungspläne im Prozeß des kommunikativen Handelns**.

Damit ergeben sich allerdings grundsätzliche Probleme für eine an der Handlungsregulation orientierte Didaktik: Welche Regeln und Kriterien, Operationsmuster und Kenntnisse allgemeiner Art sind für den Entwurf einer situationsspezifischen Handlungsstruktur notwendig, deren Entäußerung durch Einzigartigkeit gekennzeichnet ist? Wie können allgemeine Handlungsschemata im situativen Handeln verifiziert beziehungsweise falsifiziert werden, wenn das Handlungsziel erst im Handlungsprozeß gewonnen werden kann? Und beispielsweise: Inwieweit macht Projektlernen überhaupt Sinn, wenn die Projekterfahrungen nur einzigartige Projekterfahrungen bleiben können? Wo ist der archimedische Punkt, von dem aus berufliche Handlungskompetenz inhaltlich bestimmt und gelehrt werden kann?

Das Problem der Komplexität des Entsprechungsverhältnisses von inneren Handlungsplänen und äußeren Handlungen führt uns zu unserer eigentlichen Fragestellung: Wie muß die berufliche Handlungskompetenz von IT-Fachkräften beschaffen sein, die ständig vor dem Problem steht, allgemeine Regeln und Kriterien, Operationsmuster usw. auf einzigartige Projekte »anzuwenden« und ständig Gefahr läuft, mit der Fixierung auf schematisierte Handlungspläne die Gestaltungschancen in spezifischen Projekten zu zerstören?

Um diese Fragen zu beantworten, wenden wir uns wieder den drei eingangs skizzierten analytischen Gesichtspunkten zu, die wir in zwei Schritten bearbeiten werden:

- In einem ersten Schritt werden wir die einzelnen Dimensionen der beruflichen Handlungskompetenz bestimmen und bezüglich der IT-Fachkräfte konkretisieren.
- In einem zweiten Schritt soll geklärt werden, inwieweit der Handlungsspielraum, Handlungsbereich, das Handlungsraumkonzept und das Handlungsvermögen von dem historisch-gesellschaftlichen Arbeitsprozeß Software-Entwicklung **und** der beruflichen Handlungskompetenz abhängen. Gleichzeitig geht es darum, die einzelnen Dimensionen der beruflichen Handlungskompetenz bezüglich ihrer Bedeutung für die Bewältigung der Aufga-

ben **Technikgestaltung, Kooperation und Prozeßgestaltung** qualitativ zu beschreiben und aufeinander zu beziehen.

2.4.2 Drei Dimensionen der beruflichen Handlungskompetenz und die Erfordernisse der Ganzheitlichen Arbeitsgestaltungskompetenz

Die drei Dimensionen der beruflichen Handlungskompetenz

Worauf greifen wir zurück, wenn wir eine Handlungsstruktur entwerfen und eine Handlung realisieren? Was macht unsere Handlungskompetenz aus? Die zwei in Folge stehenden Fragen enthalten eine sprachliche Falle: Es wird suggeriert, daß mit der Beantwortung der ersten Frage auch die zweite beantwortet sei. Wir werden diese Falle kurzzeitig ignorieren, bis wir in sie hineingelaufen sind, um danach festzustellen, daß sie ohnehin nicht so leicht zu umgehen ist, auch wenn man sie kennt. Die Antwort auf die erste Frage läßt sich mit einer tabellarischen Aufzählung geben, in der alles aufgeführt wird, auf das wir zurückgreifen, wenn wir handeln:

- aktualisierbares Wissen[55]
- Wahrnehmung
- motorische Fähigkeiten
- Motive, Einstellungen
- soziale Verhaltensschemata.

Vorschnell könnte daraus der Schluß gefolgert werden: Die Handlungskompetenz besteht aus Wissen, Wahrnehmung, Motorik, Motivationen und sozialen Verhaltensschemata. Aber genau dies kann sich als naheliegender Irrtum entpuppen, denn: Stellen die genannten einzelnen Dimensionen tatsächlich Teile der Handlungskompetenz dar, oder ist es nicht vielmehr so, daß die Handlungskompetenz das **Zurückgreifen** auf diese Teile bezeichnet? Also: Nicht das Wissen ist entscheidend, sondern das **Aktualisieren von Wissen im Denkprozeß**, nicht das Wahrnehmen ist entscheidend, sondern der **auslösende Prozeß des Wahrnehmens** usw. Ein so gefaßter Begriff der Handlungskompetenz steht gewissermaßen als Prädikat zwischen dem handelnden Subjekt und seinen »benutzten« Objekten Wissen, Wahrnehmen usw.; er bezeichnet damit einen **subjektgesteuerten Prozeß**, der sich am besten anhand der Sprache verdeutli-

---

[55] Unbewußtes Wissen soll hier ausgeschlossen werden, da ansonsten auch Einstellungen, Motive und soziale Verhaltensweisen maßgeblich als Wissen bezeichnet werden können.

chen läßt (**ich** habe jene Regel benutzt, **ich** habe meine Aufmerksamkeit dahin gelenkt,...). Aber auch ein solcher subjektorientierter Begriff von Handlungskompetenz steht vor einem grundsätzlichen Problem, denn: Was macht das Subjekt aus, wenn Wissen, Wahrnehmen, Motivationen, Einstellungen usw. von ihm abgezogen und zu inneren Objekten deklariert werden, über die das Subjekt verfügt? Oder sollte hier generell von der Subjekthaftigkeit des Handelnden abgesehen werden, und der Entwurf einer Handlungsstruktur und seine Durchführung allein aus einem **kognitiv-emotiven Regulationsprozeß** abgeleitet werden, dessen einziger Subjektbegriff in einer Tendenz zur Herstellung eines selbstorganisierten Gleichgewichts besteht?[56] Wir glauben, daß diese Frage dahingehend verneint werden muß, daß das handelnde Individuum nicht allein aus der Regulationsperspektive betrachtet werden kann und auch nicht in einem auf ein inneres Gleichgewicht abzielenden **Prozeß der Selbstorganisation** hinreichend beschrieben wird. Vielmehr folgen wir hier Furths Weiterentwicklung des Piagetschen Denkens, der dem Menschen eine "spontane Tendenz, gesellschaftliche Beziehungen zu konstruieren" [Furth 1990, S. 190] zuschreibt. Dieser "Wunsch, sein psychisches Leben in einer gesellschaftlichen Form zu verankern" [ebd.] macht den Menschen zu einem notwendig sozialen Wesen, das wiederum auf **die Konstruktion des Subjekts als kohärente Basis von Handlungszurechnungen** angewiesen ist. Wenn ich handelnd anderen Menschen begegne, können wir uns nur als handelnde und kommunizierende Subjekte und nicht als sich selbstorganisierende Systeme begegnen, auch wenn es uns offensteht, uns als solche zu beschreiben. **Kommunikatives Handeln geht den innerlichen Regulationsprozessen der Handlungskompetenz voraus**. "Im Umgang der Individuen miteinander in der gemeinsamen Lerntätigkeit (Arbeitstätigkeit usw.) bildet sich ein entscheidender Teil der Grundlagen der Handlungskompetenz heraus. Denn nach Wygotski tritt jede psychische Funktion in der phylogenetischen und ontogenetischen Entwicklung zweimal auf, »(...) zuerst auf sozialer, dann auf psychologischer Ebene, zuerst zwischen den Menschen als interpsychische Kategorie (...), dann als intrapsychische Kategorie.«" [Höpfner 1991, S. 23].

Wir fassen zusammen: Die berufliche Handlungskompetenz besteht aus einzelnen analytisch trennbaren Dimensionen, die sich tabellarisch aufzählen lassen.

---

[56] Diesen Standpunkt vertritt z.B. Piaget in seinem Spätwerk: "Die ganze biologische Entwicklung und mit ihr zusammen die aus ihr hervorgehende Entwicklung der kognitiven Funktionen, die ganz allgemein von den permanenten Notwendigkeiten eines Gleichgewichts zwischen dem Organismus und der äußeren Umwelt (oder zwischen dem Subjekt und den Objekten) beherrscht wird, ist charakterisiert durch eine wachsende Autonomie des Organismus oder des Subjekts in ihrer Selbst-Organisation, also durch eine immer stärker verinnerlichte Äquilibration." [Piaget 1976, S. 182].

Diese Dimensionen bezeichnen aber keine statischen Kategorien, die sich zwecks des Entwurfs und der Durchführung einer Handlungsstruktur vom Individuum abrufen lassen. Vielmehr handelt es sich tendenziell um sich selbstorganisierende innerpsychische Prozesse, deren Kohärenz (Vollständigkeit des Handlungsgefüges) und Subjekthaftigkeit allein durch "inneres Sprechen" und soziale Kommunikation hergestellt werden.

Aufgrund dieser theoretischen Grundannahmen lassen sich die eingangs genannten Dimensionen beruflichen Handelns reformulieren. Wir schlagen im folgenden vor, drei analytische Dimensionen der beruflichen Handlungskompetenz zu unterscheiden[57] und auf kommunikatives Handeln zu beziehen:

- **Wissen als Prozeß** (das Zusammenspiel unterschiedlicher Wissensformen, ihrer Aktualisierung und Modifikation)
- **Soziale Kompetenzen** (das Zusammenspiel von sozialen Verhaltensschemata in komplexen beruflichen Situationen)
- **Berufliche Identität** (das Zusammenspiel von Einstellungen, Motivationen und Wahrnehmungsperspektiven)

Alle drei Dimensionen werden im Prozeß des kommunikativen Handelns auf subjekthaftes Handeln bezogen. **Erst dieser Prozeß des Kommunikativen Handelns ermöglicht den Individuen, allgemeine, verinnerlichte Schemata auf komplexes, einzigartiges Handeln zu beziehen.** Im folgenden werden wir diese drei Dimensionen näher darstellen und anhand von Handlungen von IT-Fachkräften rekursiv explizieren.

3 Wissen als Prozeß

Wissen bezeichnet sicherlich die herausragendste Dimension der beruflichen Handlungskompetenz der IT-Fachkräfte. Gleichzeitig konzentrieren sich auch die didaktischen Bemühungen in erster Linie um die Vermittlung von Wissen. Insbesondere in der Entwicklung von Curricula wird das für die berufliche Handlungskompetenz notwendige Wissen systematisch erfaßt und bezüglich seiner Handlungsrelevanz hierarchisch strukturiert. Die Problematik von **Wissen als Prozeß und handlungsregulierende Instanz**[58] wird deutlich, wenn wir uns folgendes vor Augen führen:

---

[57] Wir verzichten auf die Dimension »motorische Fähigkeiten«, da sie weder beim Entwurf noch bei der Durchführung einer Handlungsstruktur im Arbeitsprozeß Software-Entwicklung eine entscheidende Rolle spielt.

[58] Der Rückbezug von »Wissen als Prozeß und handlungsregulierende Instanz« auf Denkprozesse erscheint hier naheliegend. Da aber der eigentliche Denkprozeß sich

Das im **Gedächtnis** des Individuums »gespeicherte« Wissen ist nicht permanent »abrufbar«, sondern muß vom Individuum **aktualisiert** werden. Gleichzeitig stellt jedes begriffliche, vom Gegenstand abgezogene Wissen eine "Verallgemeinerung" [Volpert 1983, S. 9] dar, die **situationsspezifisch modifiziert** werden muß. Beide Begriffe, die **Aktualisierung** und **Modifikation**, können modellhaft vereinfacht auf **Komplexitätsskalen** beschrieben werden. Die einfachste Form der Aktualisierung findet sich in dem einfachen Automatismus des Reiz-Auslöse-Schemata: Die Ampel leuchtet rot auf, und ich aktualisiere das Wissen, bei rot stehen zu bleiben.

Die komplexen Formen der Aktualisierung bestehen in der **Vergegenwärtigung abstrakter Modelle und ihrem »selbstaktualisierenden« Prozeß (ohne äußeren Reiz) der Rekursionen und Verweisungen**: Ich habe ein didaktisches Problem und aktualisiere ein handlungstheoretisches didaktisches Modell, dessen einzelne Begriffe wiederum auf andere Modelle, z.B. das Wissensmodell, rekurrieren und gleichzeitig auf andere Modelle, z.B. das Modell des Arbeitsprozesses, verweisen. Ähnlich verhält es sich mit der situationsspezifischen Modifikation. Im einen Extremfall steht mir ein geschlossenes Handlungsschema zur Verfügung, das nur mechanistisch angewandt werden muß (und kann!): so z.B. die Befehlssyntax eines bestimmten Betriebssystems. Im anderen Fall aktualisiere ich ein offenes Handlungsschema, ein abstraktes Modell, das nur den Ausgangspunkt zur Konstruktion von Handlungseinheiten bilden kann: z.B. ein partizipatives Projektmodell der Software-Entwicklung.

Es ist naheliegend, von einer **Korrelation der Komplexitätsstufen der Aktualisierung und Modifizierung** auszugehen: Tendenziell automatisch aktualisierte Handlungsschemata werden kaum modifiziert, weil jede Modifikation einen **Prozeß der Selbstaktualisierung** in Gang setzen und damit gleichzeitig die Komplexität der Aktualisierung durch Rekursionen und Verweisungen steigern würde. Andererseits können aber auch hochkomplexe Aktualisierungen zum inneren Entwurf eines Handlungsablaufs führen, der keinerlei Ansatzpunkte zur Modifikation bietet: "Das meint nicht notwendig die Festlegung auf einen Ablauf, es kann durchaus Verzweigungen geben, sogar komplexer Art. Welche Variante aber einzuschlagen ist, ist nach klaren Wenn-Dann-Wegen festgelegt. Solche Abläufe pflegt man als »Algorithmen« zu kennzeichnen." [Volpert 1983, S. 13]. Diesem **»algorithmischen« Wissen**, welches den Großteil traditioneller IT-Ausbildungen bestimmt, steht **»heuristisches«,** bzw. **»kreatives«** Wissen ge-

---

auch den elaboriertesten wissenschaftlichen Untersuchungen entzieht, müssen wir uns auf das Beschreiben von Veränderungen des Wissens als einen Zustand des Denkens beschränken.

genüber. Auf der Ebene kognitiver Kompetenzen unterscheidet der Kognitionspsychologe Dörner analog: "Die Konstruktion von Handlungseinheiten könnte man mit Dörner der »heuristischen Kompetenz« zuordnen, während deren Auffindung der »epistemischen Kompetenz« zuzurechnen ist." [Volpert, 1983, S. 12].

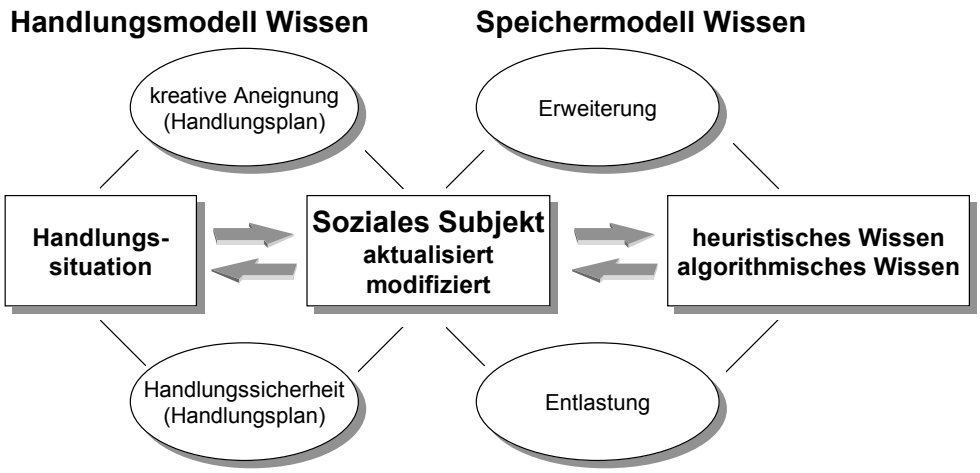
Während das heuristische Wissen den Handlungshorizont des Subjekts **erweitert** und somit der **kreativen Aneignung** eines der Handlungssituation angemessenen Handlungsplans dient, **entlastet** das algorithmische Wissen das Subjekt und verleiht seinem Handlungsplan eine der Situation angemessene **Handlungssicherheit**.[59]

Die Integration der bisher genannten Aspekte von Wissen (als wichtige Dimension der Handlungskompetenz) ermöglicht es uns, ein ganzheitliches Modell des Wissens zu entwerfen. Dieses Modell muß zwangsläufig stark vereinfachen und hat auch nicht den Anspruch, die komplexen Prozesse des Wissens in ihrer Tiefe zu erfassen. Vielmehr hat es selbst heuristischen Charakter. Sein praktischer Nutzen besteht darin, daß es sowohl die Wissensformen als auch den Wissensprozeß der IT-Aus- und Weiterbildung erfaßbar und auf die spezifischen Aufgaben des Arbeitsprozesses beziehbar macht. Da wir aus pragmatischen Gründen davon ausgehen, daß es zwischen dem Wissensprozeß sozialer Subjekte und dem didaktischen Umgang mit Wissen in der Aus- und Weiterbildung einen direkten Zusammenhang gibt, erwachsen aus dem kritischen Potential des Modells Anforderungen an pädagogische Handlungsfelder einer angemessenen IT- Aus- und Weiterbildung.

---

[59] Wir gehen im Einklang mit der Handlungsregulationstheorie davon aus, daß eine Erweiterung von Handlungsschemata und eine kreative Aneignung eines Handlungsproblems nur möglich ist, wenn genügend entlastendes Wissen zur Verfügung steht. Andererseits (und hier unterscheiden wir uns von Hacker u.a.) führt u.E. die Steigerung von entlastendem algorithmischen Wissen keineswegs automatisch zur »Freisetzung« von handlungserweiterndem Wissen und zur kreativen Aneignung einer spezifischen Handlungssituation. Das heißt: Das Beherrschen einer Programmiersprache, eines Betriebssystems, eines Daten-Funktionsmodells usw. ist notwendig, um Softwaresysteme zu entwickeln, aber keineswegs hinreichend, um spezifische Probleme eines Projektes kreativ und angemessen zu bewältigen.

Ein integriertes Modell des Wissens als »handlungsbezogener Prozeß« und als »Speichermodell«



3.1 Wissen als Prozeß und seine Bedeutung im Konzept der Ganzheitlichen Arbeitsgestaltungskompetenz

Führen wir uns jetzt den Handlungsbereich von IT-Fachkräften vor Augen, wird uns deutlich, wie einzelne Formen des Wissens und der Umgang mit Wissen das berufliche Handlungsvermögen der IT-Fachkräfte strukturieren. Den Handlungsbereich von IT-Fachkräften haben wir im ersten Teil unserer Arbeit darlegend analysiert. Der Arbeitsprozeß »Arbeitsgestaltung« läßt sich demnach in **drei einzelne, miteinander verbundene Handlungsbereiche** aufteilen:

- Technikgestaltung
- Prozeßgestaltung
- Kooperation.

Jedem dieser Handlungsbereiche ist in spezifischen Projekten ein **»objektiver« Handlungsspielraum** zugeordnet. Die Technikgestaltung ist durch das Vorhandensein bestimmter technischer Möglichkeiten begrenzt. Der Handlungsspielraum der Prozeßgestaltung ist durch zeitliche, ökonomische und in der Arbeitsorganisation zementierte Grenzen projektspezifisch limitiert. Die Kooperation ist nur im Bereich geltender sozialer Normen und vorhandener kommunikativer Fähigkeiten möglich. Die drei objektiven Handlungsspielräume bilden den Raum,

in dem das Ziel des Arbeitsprozesses, die Gestaltung von Arbeit, verfolgt werden.

Wiewohl es unbestritten ist, daß der Handlungsspielraum die objektiven Grenzen für unser Handeln markiert, so ist es auch wahr, daß unsere tatsächlichen Handlungsgrenzen vielmehr durch unser eigenes **Wahrnehmen des Handlungsspielraums** und Grenzen unseres Wissens bestimmt sind. Mit dem Begriff des **Handlungsraumkonzepts** bezeichnen wir die subjektive Wahrnehmung des Handlungsspielraums auf der einen Seite und auf der anderen Seite ihren Rückbezug auf vorhandene Qualifikationen, das heißt, in erster Linie auch Wissen.

Die Fragen lauten demnach: Welche Gestaltungsmöglichkeiten, die der Handlungsspielraum bietet, werden aufgrund welcher Qualifikationen wahrgenommen? Welche technischen Möglichkeiten sind subjektiv ergreifbar? Welche Strategien und Methoden der Prozeßgestaltung werden im Projektkontext als realisierbar erachtet? Welche Formen von Kooperation erscheinen durchführbar? Die Konstruktion des subjektiven Handlungsraums ist maßgeblich davon abhängig, welches Wissen vom Individuum aktualisiert wird und wie es modifiziert wird! Als **Handlungsvermögen** bezeichnen wir schließlich die Qualifikationen, die im Handlungsspielraum tatsächlich eingesetzt werden. Dieses berufliche Handlungsvermögen kann gewissermaßen als tatsächlich aktualisierte Teilmenge der persönlichen beruflichen Handlungskompetenz angesehen werden. Welches Handlungsvermögen mobilisiert wird, hängt allerdings keineswegs nur von dem aktualisierten Wissen ab, sondern ebenso von Faktoren wie Einstellungen, Wahrnehmungsmustern (einschließlich der Selbstwahrnehmung), Motivationen, die wir unter dem Begriff der beruflichen Identität noch näher betrachten werden, und vom Vorhandensein sozialer Kompetenzen, die, wie z.B. Kooperationsfähigkeit, nur unzureichend als bewußtes Wissen beschrieben werden können.

Wenn wir vorerst nur die Wissensdimension betrachten, so hat dies zwei Gründe: Erstens einen systematischen Grund, wie er in der Analyse der einzelnen Dimensionen der beruflichen Handlungskompetenz dargelegt wurde. Zweitens, und das wiegt schwerer, gibt es einen praktischen didaktischen Grund, der darin besteht, daß Wissen maßgeblich den pädagogischen Handlungsbereich Unterricht prägt. Im Gegensatz zu den Dimensionen berufliche Identität und soziale Kompetenzen ist Wissen leichter didaktischen Handlungen zugänglich. Der Unterricht beansprucht das Monopol für die Wissensvermittlung (trotz Praktika, in denen das Wissen nur praktisch modifiziert und durch einzelne, nicht fundamentale Aspekte ergänzt werden soll), während soziale Kompetenzen und berufliche Identität in erster Linie »naturwüchsig« sozialisiert werden. Im folgenden

werden wir die Veränderungen, die im Qualifizierungsleitbild Ganzheitliche Arbeitsgestaltungskompetenz angelegt sind, anhand der in unserem Wissensmodell entwickelten Kategorien aufzeigen.

Das Ziel des beruflichen Wissens ist es, das berufliche Handlungsvermögen zu vergrößern und adäquat den Aufgabenstellungen anzupassen. Der traditionelle Weg der IT-Aus- und Weiterbildung, dieses Problem zu lösen, besteht darin, daß vom eigentlichen Arbeitsprozeß und seinen Handlungsbereichen Technikgestaltung, Prozeßgestaltung und Kooperation abstrahiert wird. Dem Lernenden wird algorithmisches technisches Wissen und algorithmisiertes[60] Prozeßwissen (z.B. das Wasserfallmodell im Software Engineering Unterricht) vermittelt. Wie er dieses auf die Handlungsbereiche Technikgestaltung, Prozeßgestaltung und Kooperation in einem spezifischen Projekt bezieht, das heißt, in konkretes berufliches Handlungsvermögen umsetzt, bleibt dem Einzelnen überlassen. Das für die kreative Aneignung eines spezifischen Projekts notwendige heuristische Wissen wird meist ebensowenig wie der Umgang mit Wissen (seine Modifikation und Aktualisierung) problematisiert, geschweige denn inhaltlich aufbereitet. Der Versuch, das berufliche Handlungsvermögen allein durch eine auf algorithmisiertem Wissen beruhende Handlungskompetenz herzustellen, scheitert schon daran, daß der Arbeitende mit diesem Wissen kein situationsangemessenes Handlungsraumkonzept entwickeln kann.[61] Daß die einzelne IT-Fachkraft oftmals trotzdem irgendwie mehr oder weniger gut und situationsangemessen handelt, hat sie meist heuristischem, nicht bewußt eingesetztem Alltagswissen zu verdanken. Die im Zuge der betrieblichen Reorganisation von Unternehmen stattfindende »Verschmelzung« von Arbeits- und Technikgestaltung führt allerdings zu einem Anstieg der Komplexität von integrierter Technik-, Prozeßgestal-

---

[60] Der Unterschied zwischen algorithmischem Wissen und algorithmisiertem Wissen ist folgender: Algorithmisches Wissen ist sachlogisch algorithmisch (Wenn-Dann-Struktur), während algorithmisiertes Wissen Wenn-Dann-Strukturen alleine auf der Basis von Erfahrung konstruiert. Algorithmisiertes Wissen kann unproblematisch sein (wenn ich einen hochgehobenen Gegenstand loslasse, dann fällt er), aber auch problematisch (wenn mein Datenfunktionsmodell vollständig und widerspruchsfrei ist, dann habe ich alle notwendigen Daten der Systemanforderungen erhoben).

[61] Wir wollen hier ausdrücklich auf eine Differenz zwischen unserer Argumentation und der Argumentation einer Gruppe von Informatikern, die sich um ein neues Selbstverständnis ihres Faches bemühen, aufmerksam machen. Die von Floyd vertretene Position des »Software Development as Reality Construction« kommt ebenso wie Naurs programmatische Schrift "Programming as Theory Building" zu einer ganz ähnlichen Kritik algorithmisierten technischen Wissens wie wir. Der entscheidende Unterschied besteht darin, daß es ihnen um das Verstehen des **Wesens** von Software Entwicklung geht, während uns das berufliche Handlungsvermögen interessiert. Das Konvergieren beider Argumentationslinien sehen wir als Plausibilitätsindiz an [vgl. Floyd 1992a; Naur 1985].

tung und Kooperation, der nur noch in den seltensten Fällen durch heuristisches Alltagswissen bewältigbar ist. Die zweite Gefahr, die in einer Fokussierung der Aus- und Weiterbildung auf algorithmisiertes Wissen besteht, ist die: Spezifisches Handlungswissen wird verallgemeinert und algorithmisiert, obwohl es zur Erschließung von Handlungssituationen in erster Linie heuristische Bedeutung hat (so z.B. das Wasserfallmodell, aber auch graphische Datenfunktionsmodelle). Diese Verdrehung führt zu einem situationsunadäquaten Handeln in starren Handlungsschemata.[62] Eine dritte Gefahr betrifft schließlich die Aus- und Weiterbildungsinstitutionen selbst: Algorithmisches und algorithmisiertes Anwendungswissen sind oftmals gekennzeichnet durch eine extrem kurze Verfallszeit, eine Tendenz (trotz - aber auch wegen - laufender Standardisierungsbemühungen!) zur Vermehrung durch Ausdifferenzierung und durch ein potentielles Überflüssigwerden, da algorithmisches Wissen prinzipiell der Möglichkeit einer technischen Automatisierung zugänglich ist. Wenn Aus- und Weiterbildungsinstitutionen ihre Curricula allein nach dem jeweils marktgängigen algorithmischen Wissen ausrichten, geht jegliche Kohärenz der Ausbildung verloren. Sie gleicht dann zunehmend einem Flickenteppich.

Das von uns vorgeschlagene Konzept der Ganzheitlichen Arbeitsgestaltungskompetenz impliziert dagegen, Wissen prozeßhaft als Dimension der beruflichen Handlungskompetenz zu betrachten und die Fokussierung auf algorithmisiertes Wissen aufzugeben. Daraus leiten wir zwei programmatische Forderungen ab:

1. Vom algorithmisierten Wissen zum heuristischen Wissen!
2. Vom schematischen zum multiperspektivischen Wissensumgang (Aktualisierung und Modifikation)!

3.2 Vom algorithmisierten Wissen zum heuristischen Wissen

Die notwendige Reformulierung des Arbeitsprozesses Software-Entwicklung verändert die Handlungssituation der IT-Fachkräfte grundlegend. Sie verlangt von der IT-Fachkraft ein grundsätzlich anderes Problemlöse- und Entscheidungsverhalten, an dem ein allein auf algorithmisiertes Wissen fokussiertes

---

[62] Da diese starren Handlungsschemata meistens scheitern, wird im praktischen Systementwicklungsprozeß oftmals ganz auf die Anwendung formaler Methoden des Software Engineerings verzichtet. An ihre Stelle treten "simply reflect common engineering heuristics" [Lutz 1992, S. 125]. Lutz, ein Verfechter formaler Methoden, der durchaus die Notwendigkeit von Heurismen im praktischen Systementwicklungsprozeß sieht, fragt dann auch: "Specifically, what are the barriers to increasing the value of the heuristic »use formal methods« in software engineering?"[ebd.]. Die Frage müßte u.E. allerdings korrekt lauten: Warum werden formale Methoden nicht heuristisch benutzt?

Subjekt scheitern muß. Um dies zu verdeutlichen, nehmen wir einen Gedanken Dörners auf, der generell zwischen Aufgaben und Problemen, sowie verschiedenen Problemtypen unterscheidet. Aufgaben sind nach ihm dadurch gekennzeichnet, daß sowohl das Aufgabenziel als auch die Methoden zur Bewältigung der Aufgaben bekannt sind. Bei Problemen ist dies nicht der Fall.[63] Dörner klassifiziert drei Problemtypen:

Klassifizierung von Problemtypen nach Dörner

Problemtyp	**bekannt**	**unbekannt**
Interpolation	Ziel und Anzahl der Operationen	Kombination der Operationen
Syntheseproblem	Ziel	Anzahl und Kombination der Operationen
dialektisches Problem		Ziel sowie Anzahl und Kombination der Operationen

Quelle: Heeg 1988, S. 24

Die übergeordnete Aufgabenstellung der Ganzheitlichen Arbeitsgestaltung, das heißt, das Zusammenspiel der Aufgaben **Technikgestaltung, Prozeßgestaltung und Kooperation**, generiert nicht einzelne Aufgaben, deren Ziele und Lösungswege bekannt sind, sondern unterschiedliche Problemtypen. Diese können nicht allein dadurch bewältigt werden, daß ein bestimmtes algorithmisches Wissen aktualisiert und modifiziert wird, sondern sie erfordern eine kreative Aneignung, sprich, die Aktualisierung und Modifizierung heuristischen Wissens. Da sich im Arbeitsprozeß Software-Entwicklung oftmals **komplexe, dynamische und vernetzte Probleme** ergeben, muß heuristisches Wissen aus verschiedensten Bereichen aktualisiert und in einem kommunikativen Prozeß modifiziert werden. So verweisen Probleme der Anforderungsanalyse sowohl auf Probleme der technischen Gestaltung als auch auf Probleme des sozialen Prozesses

[63] Wir folgen hier Dörners Unterscheidungen, weil seine Begrifflichkeiten unsere Problemsicht verdeutlichen können. Im allgemeinen halten wir es nicht für sinnvoll, diese Differenz zwischen Aufgaben und Problemen zu machen. Sprachlich wirklichkeitsnäher ist es u.E., zwischen einfachen Aufgaben und problemgenerierenden Aufgaben zu unterscheiden.

(Interessensproblem, Kommunikationsproblem) und der weiteren Kooperationsbeziehungen. Komplexe, dynamische und vernetzte Probleme sind keineswegs durch ein einfaches Auffinden der richtigen Lösung zu bewältigen. Vielmehr ist es erforderlich, daß Probleme über einen längeren Zeitraum bearbeitet werden und daß auf die während dieses Zeitraums beobachteten Problementwicklungen reagiert wird.

Diese **inkrementelle Strategie der Problembewältigung** setzt sowohl **experimentelles Probehandeln** als auch die **Initiierung eines selbstorganisierten Lernprozesses,** das heißt, die systematische Modifikation von Wissen**,** voraus. Ansonsten besteht die Gefahr, daß die Eigendynamik der Problementwicklung sich in einem Detailproblem verliert. Aus der Notwendigkeit einer permanenten und systematischen Modifikation und Aktualisierung von Wissen durch die handelnden Subjekte folgt auch unsere zweite programmatische Forderung.

3.3 Vom schematischen zum multiperspektivischen Wissensumgang

Die Aufgabenstellung Software-Entwicklung generiert einzigartige, komplexe, dynamische und vernetzte Probleme, die nicht schematisch bewältigt werden können. Die programmatische Forderung »vom algorithmisierten Wissen zum heuristischen Wissen« nähert sich diesem Problem, muß aber u.E. noch um die Forderung nach einem multiperspektivischen Wissensumgang ergänzt werden. Die Ursache dafür liegt darin, daß die einzelnen Problemstellungen oftmals quer zu den einzelnen Handlungsbereichen Technikgestaltung, Prozeßgestaltung und Kooperation liegen, während auch das einzelne heuristische Wissen in der Regel einem dieser drei Handlungsbereiche zugeordnet ist. Der Versuch, **ein** heuristisches Wissen aufzubauen, das allen drei Handlungsbereichen gleichermaßen gerecht wird, gleicht dem naturwissenschaftlichen Glauben, eine allgemeingültige Weltformel zu finden, aus der sich alle Naturgesetze ableiten lassen. Da es keinen archimedischen Punkt gibt, von dem aus sich die Probleme der Systementwicklungsprozesse mit Hilfe von Wissen lösen lassen, besteht nur die Möglichkeit, diese Probleme multiperspektivisch zu betrachten. Als Heurismus einer multiperspektivischen Wissensorientierung schlagen wir vor, jedes komplexe Problem des Software-Entwicklungsprozesses zumindest unter den Perspektiven des Projektprozesses, der technischen Gestaltungsmöglichkeiten und der Bedeutung für die Kooperation zu betrachten. Das heißt, es wird bewußt **heuristisches Wissen verschiedener Handlungsbereiche** aktualisiert und situationsspezifisch modifiziert.[64] Integriert werden diese unterschiedlichen

[64] Diese Forderung findet sich zum Beispiel auch bei den Vertretern des Prototyping wieder, die ständig vor dem zuerst rein technischen Problem, welche unterstützen-

Perspektiven dadurch, daß sie auf das übergeordnete Ziel des Arbeitsprozesses der Software-Entwicklung, der Gestaltung von Arbeit, bezogen werden.

Die multiperspektivische Problembetrachtung führt allerdings selten zu Problemlösungen, die allen drei Handlungsbereichen gleichermaßen optimal entsprechen. Diese Zielkonflikte, die sich in der Anwendung heuristischer Modelle in der Praxis ergeben, erzwingen Kompromisse und Entscheidungen, die letztendlich nicht aufgrund von autonomen Wissensmodifikationen auf Seiten der IT-Fachkraft zu fällen sind, sondern zum Gegenstand des kommunikativen Handelns im Systementwicklungsprozeß werden müssen.

Die Notwendigkeit, die Gültigkeit und problemorientierte Angemessenheit von Wissen im kommunikativen Handeln herstellen zu müssen, erfordert vom einzelnen handelnden Subjekt Kompetenzen, die nicht mehr mit der Kategorie Wissen erfaßt werden können: Der Einzelne muß sein Wissen so darstellen, daß ihn die anderen verstehen. Um dieses zu erreichen, muß er sowohl sich selber als auch die anderen in der sozialen Situation verstehen. Dieses Verstehen des anderen und das Verstehen von sozialen Situationen, sowie das Beherrschen eines adäquaten Verhaltensrepertoires, erfordern vom Einzelnen bestimmte soziale Kompetenzen, die Thema unseres nächsten Kapitels sein werden.

4 Personengebundene soziale Kompetenzen

Die Forderung, der beruflich Handelnde müsse über soziale Kompetenzen verfügen, hat sich derart verallgemeinert, daß es geradezu als banal bezeichnet werden kann, wenn sie hier nochmals wiederholt wird. Kooperationsbereitschaft, Teamfähigkeit, Kommunikationsfähigkeit und ähnliches sind zum festen Bestandteil von Stellenanzeigen geworden. Uns geht es darum, mittels theoretischer Reflexion diese sozialen Kompetenzen sowohl bezüglich ihrer Funktionalität als auch in ihrer Genese zu analysieren, um sie so didaktisch-methodischem Handeln zugänglich zu machen.

4.1 Die Funktionalität sozialer Kompetenzen

Wir wollen radikal beginnen: Während »Wissen als Leidenschaft« äußerst ambivalente emotionale Gründe und Folgen haben kann, werden soziale Kompetenzen allgemein als Quelle des subjektiven und gesellschaftlichen Glücks an-

---

den Tools eingesetzt werden sollen, stehen: "We also examine the interplay between the software development strategy, the groups affected and the tools used" [Budde u.a. 1992, S. 80]. Auch Budde u.a. können kein heuristisches Modell nennen, das dieses multiperspektivische Zusammenspiel so miteinander verbindet, so daß eindeutige Entscheidungspräferenzen ableitbar wären.

gesehen. Dementsprechend hat eine emanzipatorisch orientierte Pädagogik den Erwerb sozialer Kompetenzen in den Mittelpunkt ihrer Erziehungsbemühungen gestellt. So verstandene soziale Kompetenzen ersetzen zugleich rigide formale, moralische Gesetze und verhelfen vernünftigem Wissen zum Durchbruch.

Dieser begriffliche Ursprung der sozialen Kompetenzen bleibt auch erhalten, wenn sie im Konzept beruflicher Handlungskompetenz verwendet werden: Soziale Kompetenzen treten an die Stelle der traditionellen Arbeitstugenden wie Fleiß, Gehorsamkeit, Stetigkeit usw. und an die Stelle der formalen Anwendung von Wissen. Im Idealfall versetzen die sozialen Kompetenzen die Individuen gleichsam in die Lage, das situativ und gesellschaftlich vernünftigste Handlungswissen intersubjektiv herzustellen und sich selbst in der Arbeit zu verwirklichen. Vereinfacht kann gesagt werden: Soziale Kompetenzen ermöglichen es dem Individuum, im kommunikativen Handeln aus individuellem Wissen intersubjektives, vernünftiges Wissen zu machen und aus beruflichen Verhaltensschemata eine berufliche Identität mit zahlreichen Bezügen zur ganzheitlichen Persönlichkeit zu gewinnen. Diese normative funktionale Sichtweise von sozialen Kompetenzen ist allerdings keineswegs so zwingend, wie sie auf den ersten Blick zu sein scheint, sondern sie geht maßgeblich auf die oben beschriebene emanzipatorische Bestimmung von sozialen Kompetenzen zurück.

Wenn auf diese normative Grundorientierung verzichtet wird, ist eine andere Sichtweise des Verhältnisses von sozialen Kompetenzen, Wissen und beruflicher Identität ebenso plausibel: Individuelles Wissen wird zwar in einem kommunikativen intersubjektiven Prozeß gewonnen, aber auch mittels sozialer Kompetenzen in einem kommunikativen Prozeß als allgemeines durchgesetzt. Als Beispiel könnte hier die soziale Kompetenz der Rhetorik angeführt werden. "Von altersher als techne ton logon bezeichnet, tritt sie überall dort auf, wo uns gesicherte Gründe des Argumentierens fehlen, Prinzipien allgemeiner und notwendiger Art nicht greifbar sind, und es um die Durchsetzung unserer Meinung geht" [Schubert 1989, S. 15]. Daß diese mittels rhetorischer Fähigkeiten durchgesetzte Meinung nicht unbedingt den Kriterien intersubjektiver Vernunft folgen muß, gilt als allgemein bekannt.

Ähnlich verhält es sich, wenn feste berufliche Verhaltensschemata zum Ausgangspunkt sozialer Kompetenzen werden, die wiederum zur kollektiven Verfestigung dieser Verhaltensschemata beitragen. Zum Beispiel läßt man den mit hohem persönlichem Engagement und Idealismus ausgestatteten Berufsanfänger systematisch scheitern, bis er sich den eigenen Verhaltensschemata fügt. Auch dieses taktische Handeln beinhaltet soziale Kompetenzen, die wir freilich in der sozialwissenschaftlichen Literatur unter dem Stichwort »soziale Kompetenzen« kaum finden werden. Wenn wir im folgenden soziale Kompetenzen als

Dimension der beruflichen Handlungsfähigkeit untersuchen, folgen auch wir der gängigen normativen Beschreibung von sozialen Kompetenzen. Diese Orientierung ergibt sich sowohl durch das Ziel des Arbeitsprozesses, Arbeit kollektiv und nicht autokratisch neu zu gestalten, als auch durch die Organisation der Arbeit in Form von Projekten, die zumindest den Anspruch haben, Probleme nicht durch Repressivität, rigide Rollenzuschreibungen und soziale Verhaltenskontrollen zu lösen, sondern durch herrschaftsfreie Kommunikation über fachliche und normative Fragen. Der Hinweis auf die »machiavellistische« Sichtweise sozialer Kompetenzen sollte aber vor einer blauäugigen pädagogischen Perspektive warnen, die den Mangel sozialer Kompetenzen bei den Auszubildenden beklagt, aber nicht sehen will, daß die Gesamtorganisation des Ausbildungssystems ganz andere »machiavellistische« soziale Kompetenzen fördert, die äußerst legitim sein können.

4.2 Eine Systematik sozialer Kompetenzen

Eine systematische Entwicklung von Grundqualifikationen des interaktionistischen Rollenhandelns hat Habermas vorgelegt. Wir werden diese Systematik und ihre Grundannahmen im folgenden kurz skizzieren, um sie in einem weiteren Schritt auf die berufliche Handlungskompetenz von IT-Fachkräften zu beziehen. Ausgangspunkt der Überlegungen von Habermas ist das Phänomen, daß sich in Interaktionssituationen **gleichermaßen Menschen und Rollenspieler** gegenüberstehen. Mit diesem Rückbezug auf Grundannahmen des symbolischen Interaktionismus wendet sich Habermas gegen die Grundtheoreme der strukturell-funktionalen Rollentheorie (Parsons), die den handelnden Menschen als mit der Rolle identisch ansehen. Als Grundqualifikationen des interaktionistischen Rollenhandelns nennt Habermas

- Frustrationstoleranz
- Ambiguitätstoleranz
- Rollendistanz
- Sprachkompetenz
- Empathie.

Diese Begriffe sollen im folgenden bezüglich ihrer Bedeutung als Dimensionen sozialer Kompetenz diskutiert werden.

4.2.1 Frustrationstoleranz

Die Frustrationstoleranz beinhaltet die Fähigkeit, die aktuelle Nichtbefriedigung eigener Bedürfnisse zu ertragen.

Während die strukturell-funktionale Rollentheorie davon ausgeht, daß in komplementären Rollen eine wechselseitige Befriedigung von Bedürfnissen eintritt (Integrationstheorem), weil mit der Rolle auch rollenspezifische Bedürfnisse sozialisiert werden, behauptet die interaktionistische Rollentheorie, daß das Ausmaß der Bedürfnisbefriedigung äußerst unterschiedlich sei. Da rollenspezifische Interaktionen auch dann aufrechterhalten werden müssen, wenn zugewiesene Rollenerwartungen nicht mit den Bedürfnissen des Subjekts übereinstimmen, ist Frustrationstoleranz eine erwünschte Grundqualifikation des Rollenhandelns. Eine kritische Würdigung des Postulats der Frustrationstoleranz kann u.E. nur erfolgen, wenn der skalierbare Charakter der sozialen Kompetenz der Frustrationstoleranz berücksichtigt und das Optimum an Frustrationstoleranz nicht mit ihrem Maximum gleichgesetzt wird: Ein Maximum an Frustrationstoleranz kommt ebenso wie ein Minimum an Frustrationstoleranz der Durchsetzung repressiver Interaktionsstrukturen entgegen.

Beispielsweise toleriert ein Beschäftigter alle ihm zugemuteten Frustrationen, weil er weiß oder glaubt, daß das Insistieren auf eigene Bedürfnisse im betrieblichen Interaktionsprozeß Repressionen zur Folge hat. Aber auch der Beschäftigte, dessen Frustrationstoleranz gegen Null strebt, entwickelt, wenn er - was wahrscheinlich ist - an seiner Arbeitsstelle verbleiben will, wirklichkeitsverzerrende kognitive Strategien (»die Aufgabe des Beschäftigten ist, zu gehorchen«), die repressiven Interaktionsstrukturen entgegenkommen. Ebenso führt auf der anderen Seite auch der Mangel an Frustrationstoleranz, wenn dem Handelnden entsprechende Machtmittel zur Verfügung stehen, zu repressivem Verhalten. Dieses Dilemma, Frustrationstoleranz als Grundqualifikation interaktionistischen Rollenhandelns zu fordern und gleichzeitig zu begrenzen, stellt sich allerdings weitaus unproblematischer dar, wenn Frustrationstoleranz mit den Grundqualifikationen Ambiguitätstoleranz, Rollendistanz, Sprachkompetenz und Empathie verknüpft wird.

4.2.2 Ambiguitätstoleranz

Ambiguitätstoleranz ist die Fähigkeit, in uneindeutigen Handlungssituationen handlungsfähig zu bleiben.

Ausgangspunkt der Forderung von Ambiguitätstoleranz ist das Phänomen, daß Rollenerwartungen und tatsächliches Handeln nie deckungsgleich sind. Rollen-

erwartungen legen nur einen Handlungshorizont fest, der von den Rolleninhabern eine situative Interpretation verlangt. Neben diese Uneindeutigkeit treten die Bedürfnisse der handelnden Subjekte, sich auch im Rollenhandeln "zugleich als unvertretbare Individuen darstellen (zu) können" [Habermas 1973, S. 126]. Auch Rollenhandeln findet in einer eher diffusen Situation statt. Als Ambiguitätstoleranz wird nun genau diese Fähigkeit bezeichnet, auch in unklaren und ambivalenten Handlungssituationen interaktionistisches Rollenhandeln zu ermöglichen, indem das Individuum die Unklarheiten und Ambivalenzen aushält. Habermas wendet sich mit dem Begriff der Ambiguitätstoleranz auch gegen die strukturfunktionale Theorie Parsons, die von einer tendenziellen Kongruenz zwischen allgemeiner Rollendefinition und subjektiver, situativer Rolleninterpretation ausgeht (Identitätstheorem) [vgl. ebd., S. 126ff]. An dieser Stelle erscheint es uns notwendig, auf ein - durch die alltagssprachliche Bedeutung bedingtes - Mißverstehen der Habermasschen Begriffe hinzuweisen: Als Ambiguitätstoleranz soll hier nur eine innere Haltung des rollenhandelnden Subjekts bezeichnet werden, sofern sie handlungsrelevant ist. Demjenigen, der die Unklarheiten und Ambivalenzen der Situation sehr wohl wahrnimmt und toleriert, aber trotzdem nach einem rigiden Rollenschema handelt, kann kaum die soziale Kompetenz der Ambiguitätstoleranz zugesprochen werden.[65] Diese Fälle sind in der Arbeitssphäre nicht so selten, wie man auf den ersten Blick annehmen könnte: Alle am Projekt beteiligten Mitarbeiter tolerieren beispielsweise die Unklarheiten und Ambivalenzen, die im gemeinsamen Rollenhandeln entstehen, und flüchten sich gleichzeitig in rigides Rollenhandeln, um die aus diesen Ambivalenzen und Unklarheiten erwachsenden Problemfelder nicht in ihren Handlungen verantworten zu müssen (Rigidität als Kalkül).

4.2.3 Rollendistanz

Die Rollendistanz bezeichnet die Fähigkeit, sich im sozialen Rollenhandeln als unverwechselbare Persönlichkeit zu behaupten.

Die bisherigen Überlegungen implizieren eine weitere Grundqualifikation interaktionistischen Rollenhandelns, die Rollendistanz. Mit dem an Goffman angelehnten Begriff der Rollendistanz wendet sich Habermas gegen eine weitere Grundannahme der strukturell-funktionalen Rollentheorie, die er als Konformitätstheorem kennzeichnet: eine "institutionalisierte Wertorientierung (Rolle) entspricht einem internalisierten Wert (Motiv) in der Weise, daß geltende Normen

---

[65] Es sei denn, der Rollenhandelnde kehrt rigides Rollenhandeln gegen die Organisation, die es einfordert. So z.B. Jaroslav Hašeks Soldat Schwejk, der die rigide und repressive Organisation des Militärs durch rigides Rollenhandeln bloßstellt.

mit hinreichender Wahrscheinlichkeit auch faktisch erfüllt werden" [Habermas 1973, S. 126].

Diese Kongruenz von sozialer Rolle und handelndem Subjekt setzt aber einen Grad von Internalisierung voraus, der nur bei wenigen Grenzfällen zutreffen dürfte. Schon allein die Tatsache, daß jedes Subjekt unterschiedliche soziale Rollen ausfüllt, verhindert die totale Internalisierung institutionalisierter Wertorientierungen, das heißt, die totale Identifikation des Ichs mit einer sozialen Rolle. Desweiteren besteht die Möglichkeit (und Notwendigkeit!) eines distanzierten reflektierenden Verhältnisses zur sozialen Rolle auch in der biographischen Integrität des Subjekts. Wie ich eine bestimmte **soziale Rolle** betrachte, hat mit meinem biographischen Hintergrund, den ich als **personale Identität** habe und erhalten will, zu tun. Rollendistanz entspringt allerdings nicht nur subjektiven Bedürfnissen, sondern wird auch sozial erwartet. Zum Beispiel: Der Mann, der seine Biographie hinter sich lassen will, verschwindet in den Bürotürmen der großen Stadt, um nur noch Sachbearbeiter zu sein. Er, der glaubte, sich durch die totale Rollenübernahme unsichtbar machen zu können, wird auffällig, bald verdächtig ... Dieses kriminalliterarische Muster, der Versuch eines Menschen, die eigene Biographie mit Hilfe der undistanzierten Rollenübernahme zum Verschwinden zu bringen, das Scheitern des Versuchs an der **eigenen Subjektivität und den sozialen Erwartungen der Interaktionspartner**, enthält anschaulich den doppelten Charakter des Rollendistanzbegriffes. Wenn wir davon ausgehen, daß Rollendistanz (bis auf wenige, fast psychopathologische Ausnahmen) generell bei den Subjekten des interaktionistischen Rollenhandelns vorhanden ist, stellt sich uns die Frage, welche **Form von Rollendistanz** von uns als optimal angesehen wird. Diese wollen wir hier kurz skizzieren, weil sie einige didaktisch-methodische Folgen für die IT-Ausbildung hat. Optimale Rollendistanz setzt u. E. voraus:

- Die **rollenspezifischen Handlungen und Interaktionen** sind dem Individuum **bekannt** und werden von ihm **beherrscht**.
- Der **Kontext**, in dem die spezifische Rolle eine funktionale Bedeutung hat, wird von dem Individuum **reflektiert**.
- Das Individuum verfügt über eine **autonome personale Identität** und kann diese **bewußt reflektierend auf seine sozialen Identitäten** (soziale Rollenübernahmen) **beziehen**.
- Die **Rollendistanz zeigt sich im Rollenhandeln** und ist keine private Nicht-Identifikation. (z.B.: Innere Kündigung).

Der unter diesen Punkten komplexeste Punkt ist das reflektierte Verhältnis von sozialer und personaler Identität. Dieses kann als **dialektisches Verhältnis im**

Bewußtsein des Handelnden beschrieben werden: "In einer Interaktion müssen wir die uns angesonnene Identität vertreten, *in einem* aber unsere persönliche Identität wahren, weil wir sonst der Gefahr einer Verdinglichung durch totales Absorbiertwerden in der sozialen Identität erliegen würden. Umgekehrt müssen wir unsere persönliche Identität zur Geltung bringen, um nicht die Gefahr des Kommunikationsverlustes und der Isolierung zu riskieren. Das bedeutet, daß wir sowohl die soziale als auch die persönliche Identität vertreten, jede aber gleichzeitig in gewissem Maße wieder zurücknehmen müssen; das heißt, daß wir unser Eingehen auf die soziale Situation nur als **»Scheinnormalität«** (...) und andererseits unsere präsentierte persönliche Identität nur als **»fiktive Einzigartigkeit«** sichtbar machen. Beide Identitäten müssen also zusammengebracht werden in einer Weise, die als Balancierung beschrieben werden kann" [Geulen 1989, S. 125 f., Herv. v. d. V.].

Spätestens mit dem Begriff der Rollendistanz wird deutlich, daß auch soziale Kompetenzen kognitive Bezüge haben. Da sich dieses soziale Wissen allerdings nicht von emotionalen und lebensweltlichen Erfahrungen trennen läßt, ist es nicht wie Unterrichtswissen vermittelbar. Wir werden darauf im Rahmen unserer Vorschläge zur Gestaltung von Aus- und Weiterbildung zurückkommen. Während die bisher dargestellten Grundqualifikationen interaktionistischen Rollenhandelns wesentlich auf das Verhältnis des einzelnen Individuums zu seinen sozialen Rollen abzielen und kontrafaktisch zu den Grundtheoremen eines verbreiteten strukturfunktionalen Rollenverständnisses ausformuliert sind, geht es bei den sozialen Kompetenzen Empathie und Sprachkompetenz um soziale Kompetenzen, die kommunikatives Handeln überhaupt erst ermöglichen.

4.2.4 Empathie

Als Empathie wollen wir das Verstehen aktueller Ausdrucksformen des in Handlungen eingebundenen Interaktionspartners bezeichnen, das auf Einfühlung beruht.

Diese Sichtweise grenzt sich von einer psychologischen Perspektive ab, die Empathie als Ausdrucksverstehen ausschließlich der aktuellen Affekte (Freude, Trauer, Zorn, usw.) ansieht, ohne deren konkreten Handlungsbezug zu berücksichtigen. Obwohl es nachweislich auch eine soziale Kompetenz Empathie gibt, die auf das Verstehen des anderen als psychologisches Objekt abzielt, wollen wir diese psychologischen Empathietheorien hier nicht weiter betrachten.[66] Vielmehr

[66] Diese psychologischen Theorien und Untersuchungen haben bisher wenig Erkenntnisse gebracht. Insbesondere gibt es keine Ergebnisse, an die pädagogisches Handeln anschließen könnte [vgl. Geulen 1989, S. 279ff.].

geht es uns darum, Empathie als **virtuellen Rollentausch** und **hermeneutisches Handlungsverstehen** zu betrachten, also dezidiert auf Situationen in Handlungszusammenhängen zu beziehen. "Verstehen eines Anderen ist geradezu als Verstehen seiner Handlungsorientierung zu definieren" [Geulen 1989, S. 286]. Erst wenn wir die **Handlungsorientierung** des Anderen verstanden haben, können wir auch dessen konkrete Handlungen und Ausdrucksformen verstehen.

Als Handlungsorientierung wird die Ziel-Mittel-Struktur des Handelns und die soziale Orientierung (z.B. Interessen) bezeichnet. Um die Handlungsorientierung des Anderen zu verstehen, müssen wir uns in ihn hineinversetzen. Diese Fähigkeit, uns in andere hineinversetzen zu können, "setzt die Verfügung über ein dezentriertes Vorstellungssystem von dem betreffenden Bereich der Wirklichkeit voraus (...)" [ebd., S 288]. Da dieses **dezentrierte Vorstellungssystem** nicht nur die Basis empathischen Handelns bildet, sondern uns gleichsam auch einen didaktischen Ansatzpunkt zur pädagogischen »Herstellung« von Empathie liefert, erscheint es uns an dieser Stelle sinnvoll, auf eine entscheidende Passage bei Geulen hinzuweisen: "Wir können die Operationen, uns in die Position eines anderen zu versetzen, erst vollziehen, wenn wir den jeweiligen Realitätsausschnitt nicht mehr ausschließlich aus der Perspektive des eigenen Standorts oder dem eines bestimmten Anderen sehen, sondern gewissermaßen aus der Sicht eines Beobachters, dem sich *beide Standorte gleichzeitig* und daher in ihrem räumlichen *Bezug zueinander* darbieten; erst unter dieser Bedingung sieht das Subjekt die Relation zwischen seinem und dem Standort des Anderen, die notwendige Voraussetzung dafür ist, von der einen Position zur anderen überzugehen." [ebd., S. 288f, Herv. i.O.].

Die Entwicklung eines dezentrierten Vorstellungssystems ist die Voraussetzung dafür, daß IT-Fachkräfte die Interessen der Anwender verstehen können.

4.2.5 Sprachkompetenz

Die Sprachkompetenz bezeichnet die Fähigkeit, sich verständlich mitzuteilen.

Die theoretische Diskussion von Sprachkompetenz stellt uns vor eine besondere Schwierigkeit: Einerseits scheint die Sprachkompetenz so selbstverständlich ein Bestandteil der sozialen Kompetenz zu sein, daß jede theoretische Ableitung überflüssig ist. Andererseits verweist das Phänomen Sprache auf einen extrem vielschichtigen Problemhorizont, dessen wissenschaftliche Diskussion zur Ausdifferenzierung einer eigenständigen Wissenschaft, der Sprachwissenschaft, geführt hat. Da selbst eine Skizze der unterschiedlichen Problemperspektiven der Sprachwissenschaft und ihrer interdisziplinären Bezüge (z.B.

Sprachphilosophie, Kommunikationswissenschaft) hier nicht zu leisten ist, beschränken wir uns auf eine Perspektive, die unserer übergeordneten Problemstellung, der Frage nach der Handlungskompetenz, entgegenkommt. Diese Einschränkung impliziert freilich nicht, daß andere Theorieperspektiven der Sprachwissenschaft für die berufliche Handlungskompetenz von IT-Fachkräften ohne Bedeutung seien.

Die von uns präferierte Theorieperspektive ist als linguistische Pragmatik bekannt geworden. Diese hat ihre Wurzeln in Wittgensteins These, daß die **Bedeutung der Sprache im Gebrauch** bestehe. Demnach geht erstens der Sinn eines Satzes über seinen expliziten, propositionalen Gehalt hinaus und zweitens geht der Sinn auf eine **Intention** zurück, die darin besteht, die Handlungen des Interaktionspartners - wie vermittelt auch immer - zu verändern. Der einfache Satz »Es regnet« kann Unterschiedlichstes bedeuten: »Laß uns nicht spazierengehen!« oder »Das Heumachen muß verschoben werden!« oder »Soll ich trotzdem noch einkaufen gehen?« oder »Ich möchte Sie, bevor wir mit unseren sachlichen Verhandlungen beginnen, zu einem Small Talk ermuntern, um eine gemeinsame, 'private' Atmosphäre zu schaffen.« Die Bedeutung des Satzes ist demnach weitgehend **situations- und kontextabhängig**. Gleichzeitig verweist sie auf etwas, das Wittgenstein als **»Sprachspiele«** bezeichnet. "Ein Sprachspiel ist eine bestimmte Weise der Operation mit bestimmten Sprachzeichen, die für einen bestimmten Typus von Situationen in der jeweiligen Lebenswelt gilt und durch Regeln - die wir in einem nicht zu engen Sinn als soziale Normen bezeichnen dürfen - festgelegt sind" [Geulen 1989, S. 319]. Die sprachpragmatische Kompetenz besteht zunächst darin, welche sprachlichen Mittel zu wählen sind, um in einer bestimmten Situation eine bestimmte Intention zu realisieren. Dies legt den einfachen Schluß nahe, daß bestimmten situationsabhängigen Rollenschemata bestimmte Sprachspiele zugeordnet sind, die bestimmte Sprechakte strukturieren: So kann man beispielsweise die unterschiedlichen Rollen im Militär betrachten, bestimmte militärische Sprachspiele identifizieren, sowie deren strukturierende Wirkung auf einzelne Sprechakte, z.B. Befehle, nachzeichnen. Die einschränkende Sichtweise dieses Vorgehens ist allerdings offensichtlich: "Es werden nur diejenigen Aspekte des Sprachverhaltens erfaßt, die sich als durch *Regeln* bzw. *Konventionen* geleitet beschreiben lassen" [Meggle 1993, S. VII]. Bei der Diskussion der Rollendistanz haben wir aber schon festgestellt, daß interaktionistisches Rollenhandeln immer auch die in den sozialen Rollen festgelegten Regeln bzw. Konventionen überschreitet. Wenn die Interaktionspartner nur Sprechakte mitteilen und verstehen können, die einem den sozialen Rollen entsprechenden Sprachspiel angehören, verfügen sie über keine Möglichkeiten, die Balance von sozialer und personaler Rolle zu kommu-

nizieren. Plastisch ausgedrückt: Die babylonische Sprachverwirrung findet sich in einzelnen Sprachspielen wieder, die sich gegenseitig nichts zu sagen haben. Eine Theorie, die an der generellen Wirkungsmächtigkeit und Rollenabhängigkeit von Sprachspielen anknüpft, hat "zu bestimmen, was es heißen soll, daß jemand mit einer Handlung, die bereits eine konventionelle Bedeutung besitzt, jemandem etwas zu verstehen zu geben (konversationell zu implizieren) versucht, was von dieser konventionellen Bedeutung nicht gedeckt ist bzw. ihr sogar widerspricht und (...) diejenigen Prinzipien anzugeben, mit Hilfe derer sich das jeweils konversationell Implizierte erschließen läßt" [Meggle 1993, S. VIII]. Eine solche Theorie fundiert die einzelnen Sprechakte nicht allein als Konvention, sondern konstruiert sie in Bezug auf eine rationale Universalpragmatik. Im folgenden skizzieren wir die Universalien, die Habermas vorgelegt hat. Ausgangspunkt ist dabei die Rekonstruktion der Charakteristik der idealen Sprechsituation, "in der die Kommunikation nicht nur durch äußere kontingente Einwirkungen, sondern auch nicht durch Zwänge behindert wird, die aus der Struktur der Kommunikation selbst sich ergeben." [Habermas 1970, S. 137]. Als grundlegende Prinzipien und sprechaktkonstruierende Selbstverpflichtungen nennt Habermas:

1. Das Prinzip **»Wahrheit«**, welches besagt, daß der kommunizierte Sprechakt auf eine äußere (egal wie vermittelte) Realität bezogen ist. Diesem Prinzip entspricht eine **Begründungsverpflichtung**.
2. Das Prinzip **»Wahrhaftigkeit«**, welches besagt, daß der kommunizierte Sprechakt die innere Realität der Intentionen des Sprechers zum Ausdruck bringen will. Diesem Prinzip entspricht eine **Bewährungsverpflichtung**.
3. Das Prinzip der **»Richtigkeit«**, welches besagt, daß sich der Sprechakt an reflexionszugänglichen Normen orientiert. Diesem Prinzip entspricht die **Rechtfertigungsverpflichtung**.
4. Das Prinzip **»Verständlichkeit«**, welches besagt, daß sich der Sprechakt an sprachlichen Normen der Verständlichkeit ausrichtet. Dieses Prinzip kann allerdings dahingehend als redundant betrachtet werden, als es mit den oben genannten Verpflichtungen impliziert wird.

Nicht jeder Geltungsanspruch muß in jedem Sprechakt thematisch werden. Das Interessante ist nun, daß Habermas diese Universalien zwar aus einer idealen Sprechsituation (= unverzerrte Kommunikation) ableitet, aber nicht als idealistisch gesetzt ansieht. "Der Vorgriff auf die ideale Sprechsituation hat für jede mögliche Kommunikation die Bedeutung eines konstitutiven Scheins" [Habermas 1971, S. 141]. Das heißt, daß wir "gar nicht kommunizieren können, ohne zu unterstellen, daß wir unverzerrt kommunizieren, auch wenn dies faktisch der Fall ist. Andernfalls könnten wir das, was wir sagen, gar nicht ernst nehmen; müßten

wir ständig zweifeln, ob wir das, was wir sagen, auch wirklich meinen, und ob wir überhaupt wissen, was wir sagen" [Peglau 1989, S. 200].

Was heißt das für unsere Fragestellung, für die Sprachkompetenz von IT-Fachkräften im Handlungsraum Software-Entwicklung? Wir behaupten folgendes:

- Dem interaktionistischen Rollenhandeln zwischen Benutzern und Entwicklern ist kein gemeinsames konventionelles Sprachspiel zugeordnet, das eine einfache Verständigung ermöglichen würde. Vielmehr tragen konventionelle, mit den einzelnen Rollen verbundene Sprachspiele dazu bei, daß die Prinzipien der idealen Sprechsituation nur verzerrt zum Zuge kommen. Die notwendige Sprachkompetenz der Rollenhandelnden besteht nun darin, verzerrte Kommunikation mittels der Reflexion der Grundprinzipien (Wahrheit, Wahrhaftigkeit, Richtigkeit und Verständlichkeit) aufzudecken und mit Hilfe von metakommunikativen Sprechakten, die diese Prinzipien situativ explizieren, zu entzerren. Diesen Teil der Sprachkompetenz, der auf die Aufdeckung und Überwindung kommunikationsverzerrender Sprachspiele abzielt, wollen wir als **analysierende Sprachkompetenz** bezeichnen.
- Das interaktionistische Rollenhandeln zwischen Benutzern und Entwicklern zielt auf die Konstruktion einer gemeinsamen Wissensbasis in einem kollektiven Lernprozeß ab. Diese gemeinsame Wissensbasis wird zwar keineswegs allein über das Medium Sprache hergestellt, sondern auch über Prototypen, graphische Modelle usw., aber sie wird immer auf Sprache rückbezogen: Es wird sprachlich über den Prototypen, über graphische Modelle usw. kommuniziert. Das heißt für uns: Die gemeinsame Wissensbasis verlangt eine gemeinsame Sprachbasis, die nur mittels des reflexiven Rückgriffs auf die genannten universalpragmatischen Sprachkriterien konstruiert werden kann. Diesen Teil der Sprachkompetenz, der auf die Herstellung einer neuen, für den weiteren Verlauf der Kommunikation verbindlichen, gemeinsamen Sprachbasis abzielt, bezeichnen wir als **konstruierende Sprachkompetenz**.

Sowohl die analysierende als auch die konstruierende Sprachkompetenz beziehen sich auf den »konstitutiven Schein« der idealen Sprechsituation und ihre universalistischen Prinzipien. Insofern kann von einer **einheitlichen Sprachkompetenz** gesprochen werden, die im Kommunikationsprozeß analysierende und konstruierende Funktionen haben kann.

Wir werden im folgenden noch näher darauf eingehen, wenn wir die Bedeutung sozialer Kompetenzen (als Einheit von Frustrationstoleranz, Ambiguitätstoleranz, Rollendistanz, Empathie und Sprachkompetenz) auf die drei Handlungsbereiche

der Arbeitsgestaltung, die Technikgestaltung, die Prozeßgestaltung und die Kooperation, beziehen.

4.3 Soziale Kompetenzen und die Handlungsbereiche der Ganzheitlichen Arbeitsgestaltung

An dieser Stelle wollen wir die Bedeutung sozialer Kompetenzen für die berufliche Handlungskompetenz von IT-Fachkräften anhand von zwei Thesen untermauern. Die erste These besagt, daß die Spezifik des Arbeitsprozesses moderner Software-Entwicklung in besonderem Maße nach hohen sozialen Kompetenzen bei den im Prozeß Beteiligten verlangt, daß soziale Kompetenzen zwar schon immer wünschenswert waren, zunehmend aber den Charakter notwendiger Kompetenzen bekommen (Globalthese). Die zweite These will dagegen ein naheliegendes Mißverständnis ausräumen, das darin bestehen könnte, daß die sozialen Kompetenzen allein dem Handlungsbereich Kooperation und abgeschwächt dem Handlungsbereich der Prozeßgestaltung zugeordnet werden, während der Handlungsbereich der Technikgestaltung praktisch auf soziale Kompetenzen verzichten könnte (spezifizierende These).

Die Plausibilität der Aussage der ersten These ist aufgrund unserer bisherigen Ausführungen evident: Das Arbeiten in Projekten, in denen die Beteiligten nicht auf feste, rollenbedingte Handlungsschemata zurückgreifen können, verlangt um so mehr an sozialen Kompetenzen, je mehr das endgültige Arbeitsergebnis nur als kollektives Arbeitsprodukt eines permanent kooperativen Arbeitsprozesses zu realisieren ist. Solange diese Kooperation punktuell beschränkt bleiben konnte, weil sich das Projektziel auf einen aus dem Arbeitsprozeß der Benutzer entkoppelbaren, formal definierbaren Arbeitsgang bezog, erschienen soziale Kompetenzen hauptsächlich zur Kooperation zwischen Entwicklern notwendig.

Diese sozialen Kompetenzen konnten aber weitgehend in feste, mit der beruflichen Rolle verbundene Handlungsschemata integriert werden. Die Frustrationstoleranz und die Ambiguitätstoleranz zwischen tüftelnden Entwicklern, die an Teilbereichen eines Programmes arbeiten, die Rollendistanz, die sich beispielsweise in einer spezifischen Witzkultur wiederfindet, aber auch die Empathie für den anderen Entwickler und die sprachkompetente Beherrschung des Sprachspiels der Fachsprache (und des Fachjargons) können als solche rollenspezifische soziale Kompetenzen von IT-Fachkräften bezeichnet werden.

Moderne Systementwicklungsprozesse, die auf eine permanente Kooperation von Entwicklern und Benutzern zurückgreifen müssen, sind mit dieser rollenspezifischen Ausprägung sozialer Kompetenzen kaum bewältigbar, vielmehr werden gerade diese eingeschliffenen Kompetenzen selbst zum Problem. Wir

glauben auch nicht, daß diese traditionelle rollenspezifische Ausprägung sozialer Kompetenzen durch neue rollenspezifische Kompetenzen ersetzt werden. Vielmehr scheint die Dynamik und projektspezifische Unterschiedlichkeit der Kooperationsbeziehungen allgemeine personengebundene soziale Kompetenzen zu erfordern, die in der jeweiligen Arbeit spezifisch realisiert werden müssen. Damit ist freilich eine Gefahr verbunden: Die Grenzen zwischen dem Berufsmenschen und dem Privatmenschen werden verwischt. Die Schutzfunktion beruflicher sozialer Handlungsschemata geht verloren, wenn sie nicht durch eine spezifische berufliche Identität wieder eingeholt wird.[67] Wir werden darauf im nächsten Kapitel zu sprechen kommen.

Wenn wir im folgenden das Verhältnis von sozialen Kompetenzen und spezifischen Handlungsbereichen der Ganzheitlichen Arbeitsgestaltung betrachten, interessieren uns insbesondere die Grenzfälle, bei denen die Notwendigkeit sozialer Kompetenzen nicht unmittelbar einsichtig ist. Dies gilt, im Gegensatz zum Handlungsbereich direkter Kooperation, für die Handlungsbereiche Technikgestaltung und Prozeßgestaltung. Wir schließen gleichzeitig die Fälle aus, bei denen die Technik und der Prozeß durch direkte zeitgleiche Kooperation gestaltet werden, da ansonsten zu Recht behauptet werden kann, daß soziale Kompetenzen nur vermittelt über den Handlungsbereich direkter Kooperation zur Gestaltung des Prozesses und der Technik beitragen. Präzise: Wir betrachten die Anteile der Arbeitszeit von Entwicklern im Software-Entwicklungsprozeß, die die Entwickler räumlich und zeitlich getrennt von den Benutzern verbringen.

Wir behaupten nun zweierlei: Die zeitlich und räumlich getrennte Entwicklerarbeit muß sich immer auf die zukünftige direkte Kooperation beziehen. Kooperation scheitert oftmals nicht am Mangel an sozialen Kompetenzen in der direkten Kooperation, sondern am Mangel an diesen Kompetenzen während der Phasen der eigenständigen Ausarbeitung. Auch beim eigenständigen Arbeiten des Entwicklers muß der Benutzer als virtuell anwesend angesehen werden. Denn: Die in der direkten Kooperation getätigten Absprachen können niemals vollständig die eigenständige Arbeit vorherbestimmen. Neue Probleme und Handlungsalternativen tauchen beim eigenständigen Arbeiten auf, deren Relevanz für die Benutzer und die zukünftigen Kooperationsbeziehungen von den Entwicklern mitberücksichtigt werden muß. Das heißt für uns: Erst das permanente Verfügen

---

[67] Dieses Phänomen wurde bisher hauptsächlich bei sozialen Berufen festgestellt und als Burn-Out-Syndrom bekannt: Das Einbringen personengebundener sozialer Kompetenzen in berufliches Handeln führt dazu, daß der Beruf die ganze Persönlichkeit (motivational, emotional, kognitiv) absorbiert, aber auch verschleißt. Nur eine berufliche Identität, die die sozialen Kompetenzen wiederum durch Professionalität beschränkt (z.B. die Empathie des Sozialarbeiters), kann dieses verhindern.

über soziale Kompetenzen ermöglicht es dem Entwickler, die Benutzer-Entwickler-Kooperationsbeziehungen kontinuierlich aufrecht zu erhalten, indem er eigenständiges Handeln auf zukünftiges, direkt kooperatives Handeln bezieht und damit auch den Prozeß gestaltet. Dieses setzt wiederum einen virtuell anwesenden Benutzer voraus. Gleichzeitig gestaltet der Entwickler auch ohne direkte Kooperation benutzerorientiert Technik, wenn die Bedürfnisse, die er dem virtuellen Benutzer unterstellt, mit den Bedürfnissen des tatsächlichen Benutzers identisch sind.[68]

Dieses »Modell des virtuellen Benutzers« ermöglicht es uns, herausragende Funktionen der einzelnen sozialen Kompetenzen im Prozeß der Software-Entwicklung zu bestimmen, ohne freilich zu behaupten, daß sich die einzelnen Kompetenzen nur auf diese Funktionen beziehen: Die **Frustrationstoleranz** wird dem Phänomen gerecht, daß der virtuelle, vom Entwickler »konstruierte« Benutzer nie identisch mit dem tatsächlichen Benutzer ist, mit dem er es in der direkten Kooperation zu tun hat. Das heißt überspitzt formuliert: Der reale Interaktionspartner »Benutzer« wird zur Quelle von Frustrationen. Die Frustrationstoleranz bezeichnet nun gerade die Balance zwischen zwei gegensätzlichen Bewältigungsmodi, die die Frustrationen durch die »Herstellung« eines identischen Benutzers minimieren wollen. Der eine Bewältigungsmodus besteht in einer intellektuellen Abwertung des realen Benutzers und kann in der Aussage »Ich weiß besser als der Benutzer selbst, was für ihn gut ist« verdichtet werden. Die direkte Kooperation wird dadurch allerdings tendenziell zur Pseudokooperation.

Der andere Bewältigungsmodus geht genau den umgekehrten Weg: Ich halte mich als Entwickler strikt an die dokumentierten Absprachen mit den Benutzern und bringe so wenig wie möglich eigene Problemlösungswege ein. Wenn das Ganze dann nicht funktioniert oder nicht optimal ist, kann ich immer sagen, daß die Benutzer nichts anderes gewollt haben. Beide Bewältigungsstrategien sind durch ein Fehlen der sozialen Kompetenz Frustrationstoleranz gekennzeichnet.

Ähnlich wie die Frustrationstoleranz läßt sich auch die **Ambiguitätstoleranz** auf den Software-Entwicklungsprozeß als sozialen Prozeß beziehen. In der traditionellen Sichtweise der Software-Entwicklung wird davon ausgegangen, daß die

---

[68] Alle für die Benutzung einer Software entscheidenden Fragen der Technikgestaltung können weder durch direkte Kooperation (Überforderung der Kooperation), noch durch den Rückgriff auf allgemeine Gestaltungskriterien (z.B.: Software-Ergonomie) gelöst werden. Vielmehr muß der Entwickler allgemeine Gestaltungskriterien auf die spezifischen Benutzerbedürfnisse beziehen, die er mit Hilfe seiner sozialen Kompetenzen konstruiert.

anfangs bestehenden globalen Probleme systematisch und kontinuierlich verkleinert werden, bis sie mittels algorithmischer Problemlöseverfahren bearbeitet werden können. Eine geringe Ambiguitätstoleranz führt dazu, daß die Verkleinerung der Probleme in algorithmisierbare Probleme möglichst schnell erfolgt, schlimmstenfalls, indem sie einfach als solche definiert werden (Arbeitsgestaltung wird z.B. auf Bildschirmgestaltung reduziert). Wir behaupten dagegen, daß der Software-Entwicklungsprozeß als sozialer Prozeß nicht nur Problemlösungen, sondern auch **Probleme globaler Art generiert**. Das heißt in letzter Konsequenz: Am Ende des Software-Entwicklungsprozesses stehen ein Softwareprodukt **und** ein Problemhorizont, der auf zukünftige Entwicklungen verweist. Dieser Überschuß an generierten Problemen, die augenblicklich oder im Rahmen des Projekts überhaupt nicht bearbeitbar sind, hat seine Ursache maßgeblich in der Kooperationsdynamik der unterschiedlichen Perspektiven von Benutzern und Entwicklern. Ambiguitätstoleranz heißt demnach, diesen Problemhorizont zu ertragen **und** konstruktiv zu bearbeiten.

Während die Frustrationstoleranz und die Ambiguitätstoleranz gewissermaßen als Kompetenzen beschrieben werden, die es dem Individuum erlauben, dem Druck komplexer Kooperations- und Problembeziehungen Stand zu halten, kann die **Rollendistanz** als personale Bedingung der - empathisch ausgedrückt - Selbstverwirklichung in diesen Beziehungen bezeichnet werden. Wie wir bei der Herleitung des Begriffs der Rollendistanz gezeigt haben, benennt die Rollendistanz diejenige Fähigkeit, Ich-Anteile in die berufliche Rolle einzubringen, ohne daß das personale Ich von der beruflichen Rolle absorbiert wird. Das heißt für den Entwickler: Ich kooperiere mit den Benutzern nicht als auswechselbarer Rollenmensch »Entwickler«, sondern als eine unverwechselbare Entwicklerpersönlichkeit. Als solche erwarte ich personale Achtung, die ich auch meinem Gegenüber zukommen lasse. Aus der Sicht des Gesamtprozesses erscheint die Rollendistanz als notwendige Bedingung der **Konfliktlösung** zwischen Entwicklern und Benutzern: Die Rollenbeziehung wird durch eine personale Beziehung gebrochen und deren Konflikte werden entschärft, ohne daß eine rein private Beziehung an ihre Stelle tritt.

Die Wichtigkeit und funktionale Bedeutung von **Empathie** erscheint so evident, daß sie hier kaum dargestellt werden muß. Nur soviel: Wir haben Empathie als Verstehen **aktueller** Ausdrucksformen des Anderen definiert. Dieses Verstehen haben wir als ein Hineinversetzen in die Handlungsorientierung des Anderen aufgrund des subjektiven Verfügens über ein dezentriertes Vorstellungssystem abgeleitet. Das heißt: Die Empathie des Entwicklers zeigt sich nicht darin, daß er über irgendwelche auf vorurteilhaftem Hineinversetzen beruhende Vorstellungen über den Benutzer verfügt, sondern nur in der direkten Kooperation mit dem

Benutzer. Auch die konstruierte Figur des virtuellen Benutzers muß sich auf tatsächlich erfahrene Ausdrucksformen des Benutzers beziehen. Empathie hat somit, ähnlich wie die Sprachkompetenz, eine konstruktivistische Komponente (die Konstruktion des Interaktionspartners als kohärente Handlungsfigur im Handlungskontext) und eine analytische Komponente (das Verstehen einzelner Ausdrucksformen). Pointiert formuliert: Über Empathie zu verfügen, besagt, daß ich über meinen Interaktionspartner **vorläufige Annahmen konstruiere**, die ich in der direkten Kooperation **überprüfe** und **korrigiere**; über Empathie nicht zu verfügen, heißt, einem mehr oder weniger bewußten Vorurteil zu folgen (beispielsweise: Benutzer wissen nie, was sie wollen, wenn man ihnen nicht sagt, was sie wollen sollen).

Auf die funktionale Bedeutung der **Sprachkompetenz** im Systementwicklungsprozeß haben wir schon im Rahmen ihrer Begriffslegung hingewiesen. Hier wollen wir deshalb dezidiert auf die Bedeutung der Sprachkompetenz beim nicht direkt kooperativen Handeln hinweisen. Vereinfacht ausgedrückt: Beim eigenständigen Entwicklerhandeln erscheint die Sprachkompetenz als Denkkompetenz, die die Geltungsbereiche von Entscheidungsfragen auf Geltungsbereiche des kommunikativen Handelns bezieht und so erst kommunizierbar macht.[69] Die Denkkompetenz nimmt die Form eines vorweggenommenen, inneren Gesprächs an. Probleme der Technikgestaltung und Prozeßgestaltung, die während des eigenständigen Handelns auftauchen, werden bezüglich ihres Geltungsbereichs (sachlich, normativ) analysiert und in eine verständliche Form gebracht (z.B. Vorlage erstellen). Darüber hinaus werden bestimmte Probleme durch den **inneren Erklärungszwang**, als den man diese Form der Denkkompetenz auch beschreiben kann, erst sichtbar: Beispielsweise hat der Entwickler ein Problem schon immer auf eine bestimmte Art gelöst und merkt erst beim Versuch des inneren Erklärens, daß das ihm bisher als selbstverständlich Erscheinende auch ganz anders gelöst werden kann.

Zusammenfassend kann gesagt werden: Wenn wir von dem Software-Entwicklungsprozeß als sozialem Prozeß sprechen, setzen wir die direkte Entwickler-

---

[69] Piaget und seine Mitarbeiter haben in verschiedenen Versuchen nachgewiesen, daß die Fähigkeit zu denken nicht von der Sprachkompetenz ursächlich abhängt [vgl. Piaget 1981, S. 176ff.]. Piagets Mitarbeiter Sinclair sieht die Funktion der Sprache für die Denkkompetenz dann darin: "Die Leistung der Sprache muß daher auf einer anderen Ebene gesucht werden. Die Sprache kann die Aufmerksamkeit auf die zu einem Problem gehörenden Faktoren lenken, ebenso wie sie perzeptuelle Aktivitäten zu kontrollieren vermag, wie Luria und seine Mitarbeiter nachgewiesen haben: Sprache kann somit Operationen vorbereiten, doch für die Bildung konkreter Operationen ist sie weder zureichend noch notwendig." [Sinclair zit.n. Furth 1981, S. 190]. Nur im Sinne **fokussierter Aufmerksamkeit** sprechen wir von Sprach- als Denkkompetenz.

Benutzer-Kooperation als notwendig voraus, ohne allerdings den sozialen Prozeß auf diese direkte Kooperation zu begrenzen. Insbesondere wollen wir hier abschließend auf die hohe Bedeutung sozialer Kompetenzen für Arbeitsphasen des sozialen Prozesses Software-Entwicklung hinweisen, die auf den ersten Blick jeden sozial-kooperativen Zusammenhang vermissen lassen.

Mit diesen Überlegungen schließen wir die Diskussion der sozialen Kompetenzen als Dimension beruflicher Handlungskompetenz ab, um uns einem Themenkomplex zuzuwenden, der im allgemeinen wenig Beachtung findet und doch maßgeblich mit der beruflichen Handlungskompetenz verbunden ist. Wir haben schon weiter oben angesprochen, daß soziale Kompetenzen immer an eine berufliche Identität gekoppelt sind, die diesen eine bestimmte, oftmals einschränkende Ausprägung verleiht. Die Rigidität traditioneller beruflicher Identitäten scheint kaum eine Basis für die Entwicklung adäquater sozialer Kompetenzen zu bieten, wie sie zur Bewältigung des Arbeitsprozesses Software-Entwicklung notwendig sind.

5 Berufliche Identität und Handlungskompetenz

Daß die berufliche Handlungskompetenz von IT-Fachkräften Wissen und soziale Kompetenzen voraussetzt, muß kaum begründet werden. Es ist eher die Selbstverständlichkeit dieser Kategorien, die einer angemessenen Problemsicht im Weg steht und eine theoretische Durchdringung erforderlich macht. Der Begriff der beruflichen Identität ist dagegen als Dimension der beruflichen Handlungskompetenz im alltäglichen Denken wenig verbreitet. Während sich Wissen und soziale Kompetenzen im Handeln derart widerspiegeln, daß sie als Teilkompetenzen einzelner Handlungen identifiziert und bewertet werden können, ist dies bei der Dimension beruflicher Identität kaum möglich. Identität scheint von außen kaum sichtbar zu sein und erst dann zum Problem der Handlungskompetenz zu werden, wenn sie eine dysfunktionale Form annimmt. Beschädigte Identität, gebrochene Identität, Identitätsverlust und Identitätsstörung sind gängige Begriffe, die dem sozial auffällig gewordenen Individuum zugeschrieben werden. Wenn die Beschädigung und der Verlust von Identität mit einer eingeschränkten, der Situation unangemessenen Handlungskompetenz einhergehen, wird u.E. auch die Frage aufgeworfen, inwieweit die Handlungskompetenz des Individuums von seiner jeweiligen Identität abhängig ist.

5.1 Drei Aspekte der Identität

Eine funktional auf Handlungskompetenz bezogene Identität kann definiert werden als "ein System von Regeln und Interpretationsmustern, mittels derer das

Handlungssubjekt die ihm entgegentretenden Ereignisse ordnen und ihnen gegenüber eine Autonomie entfalten kann" [Schmitz zit.n. Mader 1985, S. 37]. Diese Definition ist allerdings, worauf Mader zu Recht hinweist, dahingehend zu weit gefaßt, als sie auch jedes psychische Wahnsystem, das intersubjektiv nicht verstehbar ist, mit einschließt. Er empfiehlt daher diese Definition um eine interaktionistische Komponente zu ergänzen, die Erikson schon 1950 als entwicklungspsychologische Kategorie[70] eingeführt hat. Identität ist demnach, "die angesammelte Zuversicht des Individuums, daß der inneren Gleichheit und Kontinuität die Gleichheit und Kontinuität seines Wesens in den Augen anderer entspricht" [Erikson zit.n. Mader 1985, S. 38]. Sobald sich die Identität auf den Austausch mit anderen Menschen bezieht, bezieht sie sich immer auch auf eine äußere Objektwelt. Das heißt: Identität kann sich nur als Gleichheit und Kontinuität des Selbst herstellen, wenn sie über eine kommunizierbare kohärente »Weltsicht« verfügt. »Selbstsicht« und »Weltsicht« sind gleichermaßen verschränkte Dimensionen der Identität. Daher verweist diese Interaktionsperspektive immer auch über sich selbst als Interaktionssystem hinaus: "Für die Identität ist der Realitätsbezug oder Realitätsverlust entscheidend" [Mader 1985, S. 38]. Wenn der Verlust von Realität Identitätskrisen bewirkt, wird vorausgesetzt, daß Realität jenseits der Identität existiert und Identität sich erst in dieser Entgegensetzung strukturiert. Gleichzeitig behaupten wir, daß diese gesellschaftliche Realität sich nicht aus einzelnen Interaktionssystemen ableiten läßt, sondern jenseits von diesen existiert und auf diese einwirkt. Bevor wir dies beispielhaft illustrieren werden, wollen wir das bisher Diskutierte zusammenfassen:

- Identität kann als **subjektive Folie des Wahrnehmens und Erkennens** bezeichnet werden. Diese Folie schließt eine kohärente Selbstsicht mit ein. Das heißt: Das Individuum begrenzt die möglichen Wahrnehmungen und Wissensverknüpfungen, indem es sich selbst als Identisches (mit **einer** Biographie und einer Ich-Identität trotz vieler Rollenidentitäten) entwirft und so seine autonome Handlungsfähigkeit erhält. **Wir nennen diesen Aspekt von Identität den psychologischen Aspekt von Identität.** Dieser bezieht sich gleichsam auf die kognitiven Strukturen des Individuums, sowie auf das Phänomen des psychischen "Verlangens nach individueller Konsistenzsicherung" [Kohlstruck 1990, S. 68].
- Diese Identität kann sich nur in der sozialen Interaktion herausbilden, indem sie sich - das heißt: ihre Handlungsfolgen - interaktiv bewährt und interaktiv

[70] Wir nennen hier Erikson, der Identität als psychisches Äquivalent von soziologisch beschriebenen Interaktionsprozessen bestimmt, bei denen die Rollenübernahme durch das spontane Ich gebrochen wird [vgl. Erikson 1966].

bestätigt wird. Identität ist gleichzeitig Bedingung und Folge von interaktionistischem Rollenhandeln.[71] Daß sich dieses Rollenhandeln auf eine Balance von sozialer und personaler Identität beziehen muß, haben wir in dem Abschnitt zur Rollendistanz herausgearbeitet. Krappmann bezeichnet diese Balance von sozialer und personaler Identität als Ich-Identität.[72] Diese Ich-Identität ist sowohl Grundlage als auch Folge von Interaktionsvorgängen. **Wir nennen diesen Aspekt von Identität den interaktionistischen Aspekt von Identität**.

- Identität bezeichnet immer einen bestimmten Realitätsbezug, ohne daß Identität diese Realität determiniert. Das heißt: Die subjektive Folie des Wahrnehmens und Erkennens kann mit einer gesellschaftlich konstruierten Realität in Konflikt geraten, einem Konflikt, bei dem, überspitzt formuliert, die Identität immer den kürzeren zieht. Gleichzeitig kann diese Realität auch nicht aus den Interaktionsbeziehungen von Interaktionssystemen abgeleitet werden. Als materielle und gesellschaftlich konstruierte Realität tritt sie als Äußeres sowohl dem Interaktionssystem als auch dem psychischen System des Individuums gegenüber. **In diesem Sinne sprechen wir vom Realitätsbezug der Identität**.

Wir wollen diese dreifache Sicht von Identität an einem alltäglichen Beispiel illustrieren: Wir betrachten einen Familienvater, der über eine sehr traditionell geprägte Rollenidentität verfügt, das heißt, eine bestimmte Wertorientierung und spezifische Lebensziele verinnerlicht hat, die seine Handlungen strukturieren (psychologischer Aspekt der Identität). Diese Handlungsorientierung besteht aus traditionellen Werten, die die familiären Interaktionsformen wie -inhalte bestimmen, aber gleichzeitig selbst einen interaktionistischen Ursprung haben (interaktionistischer Aspekt der Identität). Die familiäre Interaktion bestätigt ihm

---

[71] Identität "ist **ausschließlich** konzipiert (...) als Konstituens und Folge von Interaktion" [Kohlstruck 1990, S. 66]. Die Problematik dieser dialektischen Sicht von Identität besteht darin, daß sie sowohl hinsichtlich der psychologischen Fundierung, woher denn das Verlangen nach individueller Konsistenzsicherung kommt, als auch hinsichtlich komplexer gesellschaftlicher Realität jenseits des interaktionistischen Rollenhandelns blind ist [vgl. ebd., S. 66ff.].

[72] "Letztendlich wird mit Ich-Identität die Form gemeint, unter der Akteure, die in verschiedenen Interaktionen stehen und standen, ein Interaktionssystem exekutieren und (innovativ) reproduzieren" [Kohlstruck 1990, S. 70]. Mead, der diese Differenz von Rollenidentität und Ich-Identität mit den prozeßhaften Begriffen »me« (Rollenübernahme) und »I« (spontane Individualität im Rollenhandeln) erstmals kennzeichnete, steht ebenso wie seine Nachfolger (Goffman, Habermas, Krappmann u.a.) vor dem Problem, die Ich-Identität zwar sozial verorten, aber nicht sozial bestimmen zu können. Es bleibt beim »irgendwie« [Eine Darstellung und Kritik dieser Problematik findet sich z.B. bei Kohlstruck 1990, S. 59ff.; Geulen 1989, S. 108ff.].

wiederum seine Identität. Kinder gehorchen und verhalten sich respektvoll, die Frau führt den Haushalt usw., kurz, die Rollenidentität wird in der Familie bestätigt. Die gesellschaftliche Realität wird hier anscheinend weitgehend aus dem Interaktionssystem Familie ausgeklammert, und trotzdem ist sie es, die Identitätskrisen auslöst und das Interaktionssystem gefährdet. Die gesellschaftliche Realität hat sich verändert (Wertewandel) und ist nicht mehr mit diesem Interaktionssystem Familie und den dazugehörigen Identitäten verknüpfbar. Die gesellschaftliche Reproduktionsleistung dieser Familie wird in Frage gestellt und auch faktisch zunehmend problematisch, das heißt: die Kinder werden einerseits in der Schule als gehemmt und autoritätsgeschädigt wahrgenommen, andererseits finden sie sich in anderen Interaktionskontexten tatsächlich nicht mehr zurecht (interaktionsübergreifender Realitätsbezug der Identität).

Wenn gesellschaftlich konstruierte Realität selbst die Identitäten und Interaktionen eines extrem abgeschotteten Interaktionssystems wie der Familie letztendlich vermittelt trifft, dann erscheint es einleuchtend, daß gesellschaftliche Realitätsbezüge strukturierend und destrukturierend auf systemisch hochintegrierte berufliche Interaktionssysteme wirken.

Bevor wir uns den speziellen Problemen der beruflichen Identität von IT-Fachkräften zuwenden und ihre didaktische Bedeutung darlegen, soll zuvor das Verhältnis von beruflicher Identität und anderen Identitätsformen geklärt werden.

5.2 Rollenidentität, personale Identität, Ich-Identität und berufliche Identität

Eine Darstellung der unterschiedlichen Bedeutungen des Identitätsbegriffs in verschiedenen Theorie- und Fachbezügen ist hier ebensowenig zu leisten, wie eine Darstellung der Differenzierungen, die der Begriff durch adjektivische Zusätze selbst erfahren hat. Die Strukturierung des semantischen Feldes »Identität« durch die von uns eingeführten drei Aspekte von Identität orientiert sich deshalb auch nicht an dem universalistischen Anspruch, alle Identitätsphänomene systematisch zu erfassen, sondern an der spezifischen Aufgabe, Identität als Teil der beruflichen Handlungskompetenz zu bestimmen und als didaktische Kategorie verfügbar zu machen. Wenn wir hier die Spezifik der beruflichen Identität in der Entgegensetzung zu anderen Identitätsbegriffen herausarbeiten, so geschieht dies zum einen, um die Relevanz des Begriffes darzulegen und naheliegende Mißverständnisse zu vermeiden, zum anderen, um bestimmte didaktische Ansatzpunkte zu gewinnen. Dazu möchten wir drei Thesen formulieren, die wir kurz begründen werden:

1. **Berufliche Identität ist der Rollenidentität übergeordnet**. Der in einem spezifischen Funktionszusammenhang Arbeitende füllt eine bestimmte betriebliche Rolle aus, die keineswegs mit seiner beruflichen Identität identisch ist. Die betriebliche Rollenidentität eines Schreiners in einer hochgradig arbeitsteilig organisierten Möbelfabrik hat beispielsweise wenig Gemeinsames mit der betrieblichen Rollenidentität eines Schreiners in einem kleinen spezialisierten Handwerksbetriebs. Selbst innerhalb eines Betriebes sind die betrieblichen Rollen einer Berufsgruppe äußerst unterschiedlich. Trotzdem verfügen sie über eine in vielen Punkten ähnliche berufliche Identität, die auch handlungsorientierend wirkt. Das heißt: Die beruflichen Handlungsorientierungen der Arbeitenden, Selbstwertgefühl, Motivations-, Wahrnehmungs- und Interaktionsstrukturen werden maßgeblich auch von der beruflichen Identität bestimmt.

   Wir wollen dies an zwei Beispielen, der Übernahme von Verantwortungen und Verantwortlichkeiten bezüglich von Arbeitsaufgaben und der Strukturierung informeller Kooperationsbeziehungen in der Arbeit einerseits, der Restrukturierung betrieblichen Interaktionsverhaltens durch die berufliche Identität andererseits, verdeutlichen. Mit der beruflichen Identität ist die sozial erwartete Übernahme von einer Verantwortung für ein bestimmtes Aufgabenspektrum verbunden. Gleichzeitig sind auch an die betriebliche Rolle spezifische Erwartungen von Verantwortungen geknüpft. Bei Nichtkongruenz dieser Verantwortungserwartungen kommt es dazu, daß die betriebliche Rollenerwartung vom Handelnden unter- bzw. überschritten wird: Der Schreiner in der Möbelfabrik macht Verbesserungsvorschläge, die das gesamte Produkt betreffen, obwohl seine betriebliche Rollenverantwortung nur als Zuverlässigkeit beim Holzzuschnitt definiert ist. Der technische Ingenieur übernimmt in erster Linie nur die Verantwortung für die technische Funktionsfähigkeit der Produkte, obwohl er die betriebliche Rolle eines Betriebsleiters innehat und somit auch mit betriebs- und personalwirtschaftlichen Verantwortungserwartungen konfrontiert wird. Ebenso werden beispielsweise auch die mit der betrieblichen Rolle verbundenen Interaktionserwartungen von Interaktionsstrukturen überlagert, die durch die berufliche Identität strukturiert sind. Werkstattmeister und Ingenieure interagieren z.B. auch als »Techniker« und nicht nur in ihren komplementären betrieblichen Rollen als Planende bzw. Ausführende.[73]

---

[73] Hierzu ein eindrückliches historisches Beispiel: Der straffe Durchsetzungsversuch strikten betrieblichen Rollenverhaltens um die Jahrhundertwende gipfelte in der "direktorale(n) Verfügung, die dem Konstrukteur das Betreten der Werkstätten überhaupt nicht gestattet." [Neef 1982, S. 98]. Diese Trennung widersprach allerdings

Die betriebliche Rollenidentität wird durch die berufliche Identität gebrochen. Das Ausmaß der **Kompatibilität von beruflicher Identität und betrieblicher Rollenidentität** entscheidet neben anderen Faktoren darüber, wie dieser Bruch vom Individuum realisiert wird.

2. **Berufliche Identität bildet einen Teil der personalen Identität.** Als personale Identität haben wir bisher die konsistente Biographie (Habermas) und den in verschiedenen Rollen identischen Anteil (Goffman) eines Individuums bezeichnet. Wenn wir nun behaupten, daß berufliche Identität einen Teil der personalen Identität bildet, so haben wir zweierlei vor Augen: erstens den **äußeren Zwang**, sich als Berufsmensch selbst definieren zu müssen, zweitens das **innere Bedürfnis**, sich eben das zur eigenen Sache zu machen, was man tun muß.

   Die erste These hat Max Weber prägnant mit dem Satz "Der Puritaner wollte Berufsmensch sein, - wir müssen es sein" zum Ausdruck gebracht. Dieser äußere Zwang, sich über den Beruf definieren zu müssen, gilt sogar dann, wenn das Individuum ihn nicht in der Arbeit realisieren kann. Entgegen einer in den 80er Jahren vorherrschenden Diskussion um das Ende der Arbeitsgesellschaft, das dazu führen würde, daß sich "persönliche Identität aus der Berufsrolle herauszulösen (beginne)" [Beck zit.n. Baethge 1991, S. 10], stellt Baethge fest: "Für große Gruppen von Beschäftigten, nicht zuletzt für Frauen, scheint heute eher eine entgegengesetzte Dynamik zu gelten: daß die Berufsrolle eine integrale Funktion für die persönliche Identitätskonstruktion wie für deren Stabilisierung gewinnt, bzw. weiterhin hat" [ebd.]. Gerade die berufliche Rolle - und nicht die scheinbar realere betriebliche Rolle(!) - wird zum Bezugspunkt personaler Identität. Und damit kommen wir zur zweiten These, dem individuellen Bedürfnis, berufliche Identität mit der personalen Identität zu verbinden.

   Der Aufbau einer beruflichen Qualifikation ist mit einer immensen Investition auf Seiten des Individuums verbunden. Diese psychische Investition strukturiert biographische Zeit, die als besonders prägend angesehen wird (Jugendzeit und junges Erwachsenenalter). Auf der anderen Seite verlieren lebensweltliche Zusammenhänge zunehmend an identitätsstiftender Bedeutung: "Wo zentrale Bereiche der Lebenswelt von Sinnentzug bedroht sind

---

fundamental der damaligen beruflichen Identität von Ingenieuren, die sich auf theoretische Planung und praktische Ausführung von Technik bezog. Die in vielen Unternehmen vollzogene Rücknahme dieser interaktionsbeschränkenden Rollendefinition war keineswegs in erster Linie den Sachzwängen des Arbeitsprozesses geschuldet, sondern den Widerstandshandlungen der um ihre vorgefaßte berufliche Identität kämpfenden Ingenieure [vgl. ebd.].

(z.B. Religion, Familie), ist der Ausweg, die Arbeit mit Sinn zu befrachten, so sinnlos nicht" [ebd., S. 17]. Das heißt allerdings auch, daß die in der beruflichen Identität sich widerspiegelnden Handlungsorientierungen auch mit lebensweltlichen Bedürfnissen verquickt werden, die traditionell der personalen Identität zugerechnet werden, so z.B. mit allgemeinen Wertorientierungen und Lebenszielen. Baethge spricht in diesem Zusammenhang von einer "zunehmenden normativen Subjektivierung der Arbeit" [ebd., S. 6]. Die steigende freiwillige und unfreiwillige Mobilität der Arbeitenden verhindert, daß sie ihre personale Arbeitsidentität an einer bestimmten betrieblichen Rolle orientieren können, und zwingt sie gleichsam dazu, diese an eine weitgehend im Ausbildungssystem angenommene berufliche Identität anzubinden.

Zusammenfassend können wir feststellen: Berufliche Identität und personale Identität stehen in einem Wechselverhältnis. Dieses Wechselverhältnis spiegelt die Dialektik von gesellschaftlichem Zwang (und Kontingenz) und innerem Bedürfnis wider, hat also eine doppelte Geschichte, eine gesellschaftliche und eine biographische.

3. **Die qualitative Ausprägung der beruflichen Identität markiert Grenzen und Möglichkeiten der Ich-Identität**.

Unter qualitativer Ausprägung der beruflichen Identität verstehen wir: Erstens, die Stärke ihrer Verankerung in der personalen Identität, zweitens, die Kohärenz der beruflichen Identität als kohärente Weltsicht, drittens, die Stabilisierung dieser kohärenten Weltsicht durch eine berufliche Gruppe. Um diese These zu begründen, müssen wir zuerst den Begriff der Ich-Identität kritisch diskutieren. Dieser stellt uns vor besondere Schwierigkeiten, die wir in zwei Schritten aufzeigen wollen:

- Der Begriff der Ich-Identität in der Tradition des symbolischen Interaktionismus (Mead) charakterisiert eine **spezifische Interaktionssituation** des Rollenhandelns, bei der das handelnde Subjekt die **Balance zwischen Rollenidentität und personaler Identität** herstellt und sozial erwartet herstellen muß. So gesehen bezeichnet der Begriff nichts weiter als eine dem Individuum äußerliche Situation [vgl. Kohlstruck 1990, S. 68]. Da die Theoretiker dieser Ich-Identität keinen plausiblen innerpsychischen Regulationsmechanismus angeben, der diese Balance herstellt, muß von einem dritten »Akteur« ausgegangen werden, der dieses Gleichgewicht aktiv anstrebt und anstreben kann. "Der Begriff der »Balance« zwischen persönlicher und sozialer Identität ebenso wie das dialektische Schema sind deskriptive, modellartige Kategorien für diese

Situation, treffen aber noch nicht das Phänomen der Ich-Identität als solches: »Identität« ist uns nicht *als* »Balance« gegeben" [Geulen 1989, S. 126f]. Da wir das Vorhandensein von Ich-Identität als phänomenologischen Tatbestand ansehen, können wir der Frage nach der regulierenden Instanz auch nicht dahingehend ausweichen, daß wir sie ins Reich der nur theoretisch relevanten Spekulationen verweisen. Alltagssprachlich ausgedrückt: Wir können wahrnehmen, inwieweit wir entsprechend den uns angesonnenen Rollenerwartungen handeln und inwieweit wir im Rollenhandeln unsere biographischen Vorerfahrungen einfließen lassen. Aber sobald wir diese Phänomene wahrnehmen, sehen wir auch, daß sich unser Handeln nur dadurch strukturiert, daß **wir** diese Identitäten aufeinander beziehen. Während die personale Identität und die Rollenidentität etwas sind, das wir **haben**, bezeichnet die Ich-Identität dieses **»wir«** als etwas, das wir untrennbar **sind**.

- Da wir über keine Ich-Identität in der Gestalt reformulierbarer Verhaltenserwartungen verfügen, liegt es erst einmal nahe, den Begriff der Ich-Identität von außen einzukreisen. Den ersten Schritt der funktionalen Bestimmung von Ich-Identität als Rollenbalance im interaktionistischen Handeln haben wir bereits beschrieben. Den zweiten Schritt hat Habermas gemacht, indem er diese für die Interaktion notwendige Funktion der **Ich-Identität als Qualifikationsmerkmale des einzelnen Individuums** reformulierte. Die im vorstehenden Kapitel ausführlich dargestellten sozialen Kompetenzen bezeichnet Habermas synonym als Ich-Identität [vgl. Tillmann 1989, S. 214]. Ich-Identität ist demnach keine Instanz (als prinzipiell unveränderliche Substanz) des handelnden Subjekts, sondern sie markiert qualitative Regulationsprozesse des Subjekts in ihrer Prozeßhaftigkeit. Damit ist unsere Frage nach dem regulierenden, aber selbst identisch und damit identifizierbar bleibenden Akteur (Instanz) allerdings nicht beantwortet, sondern nur in das Innere des handelnden Subjekts verschoben worden.

  Geulen schlägt nun vor, diese Leerstelle mit der **subjektiven Identität des selbstreflektierenden Subjekts** zu besetzen: "In einer sozialen Interaktion bin ich mir dadurch als »identischer« gegeben, daß ich die angesonnene soziale Identität zusammenbringe mit meiner biographisch bestimmten persönlichen Identität und mich gleichzeitig als das Subjekt weiß, in dem diese heterogenen Korrelate zusammenkommen. Erst hierdurch wird Identität in der Zeit und in verschiedenen Situationen möglich; erst das Subjekt, das sich *als* Subjekt einer solchen Synthese weiß, ist *sich* in Zeit und sozialem Raum auch »identisches«" [Geulen

1989, S. 127f]. Diese Bestimmung von Ich-Identität ist keineswegs nur theoretisch relevant: In der Selbstreflexion des Subjekts werden die Grenzen und Möglichkeiten der Welt- und Selbstsicht, wie sie durch die berufliche Identität markiert sind, erst als solche deutlich und damit auch didaktisch anschlußfähig.

Zusammenfassend können wir feststellen: Ich-Identität ist die in der Sozialisation hergestellte **Vermittlung** von der **Rollenbalance** als funktional notwendige Interaktionserwartung, von **sozialen Kompetenzen** des Individuums und der **Selbstreflexivität des Subjekts**.[74]

Wenn wir von der beruflichen Identität als markierte Grenzen und Möglichkeiten der Ich-Identität sprechen, meinen wir daher folgendes:

1. Die Rollenbalance zwischen personaler Identität und betrieblicher Rollenidentität wird im beruflichen Handeln durch eine Ich-Identität hergestellt, die durch die berufliche Identität gebrochen ist.
2. Soziale Kompetenzen sind zwar unveräußerlicher Anteil der Ich-Identität, aber sie erhalten im beruflichen Handeln über die berufliche Identität eine bestimmte Ausprägung.
3. Die Selbstreflexivität des Subjekts kennt nicht nur ein identisches Selbst, sondern als **ein Moment von diesem** erscheint das berufliche Selbst, das wir mit dem Begriff der beruflichen Identität bezeichnen.

Kurz zusammengefaßt: Während die drei anfangs vorgestellten Aspekte der Identität uns Hinweise auf die **»vermittelten Orte von Identität«** im allgemeinen (**psychischer Apparat, soziale Interaktion und Realität**) gegeben haben, haben wir nun das Verhältnis der beruflichen Identität zu den spezifischen **handlungsrelevanten Identitätsformen (der Rollenidentität, der personal-biographischen Identität und der Ich-Identität)** dargelegt und somit die **»handlungsrelevante Dynamik von Identitätsverhältnissen«** betrachtet. In der Realität finden wir freilich nicht einmal »Orte der Identität« und ein anderes Mal »dynamische Identitätsverhältnisse«. Vielmehr dienen beide Begriffe als heuristisches Modell, um sich einer komplexen beruflichen Handlungswirklichkeit begrifflich zu nähern. Da diese Wirklichkeit dem didaktischen Handeln als un-

---

[74] Dieser Zusammenhang ist von Döbert/Nunner-Winkler auch empirisch belegt nachgezeichnet worden [Döbert/ Nunner-Winkler 1975].

vermitteltes Tätigkeitsmodell in der Regel[75] nicht vorliegt, sind derartige Modelle für den Didaktiker unverzichtbar.

Im folgenden wollen wir die spezielle Ausformung der beruflichen Identität von IT-Fachkräften, wie sie sich uns in zahlreichen Gesprächen mit Praktikern dargestellt hat, untersuchen und auf die von uns erarbeiteten theoretischen Aspekte beziehen. Anschließend werden wir die Genese von beruflicher Identität aus beruflichen Vorerfahrungen und Vorerfahrungen in der Ausbildung betrachten, um so zu didaktischen Fragestellungen überzuleiten.

5.3 Die berufliche Identität von IT-Fachkräften

Die betrieblichen Rollen der IT-Fachkräfte sind so diffus, daß zu Recht von einer Berufsgruppe ohne Berufsbild gesprochen werden kann [vgl. Boß/ Roth 1992]. Wie wir aufgezeigt haben, wäre es allerdings ein Trugschluß, aus der Diffusität der betrieblichen Rollen auf eine generelle Diffusität der beruflichen Identität zu schließen. Vielmehr wird gerade innerhalb der Ausbildung, die von dem einzelnen eine hohe psychische Investition verlangt, eine berufliche Identität als Teil der personalen Identität aufgebaut und durch Gruppenprozesse (z.B. in der Fachabteilung oder auch nur mit einem Kollegen) stabilisiert. Dies mag weniger für IT-Fachkräfte zutreffen, die aufgrund eines, auch als solchen empfundenen äußeren Zwangs, wie z.B. einer Umschulungsmaßnahme, diesen Beruf ergriffen haben. Aber nicht zuletzt die zunehmende Organisierung der Ausbildung von IT-Fachkräften als originäre Ausbildung führt zur sozialen Formierung einer beruflichen Identität, der sich auf Dauer auch die Berufsnebeneinsteiger nur auf Kosten von partieller Ausgrenzung widersetzen dürften. Gerade die Disparität der betrieblichen Rollen, die im Zuge einer extrem hohen beruflichen Mobilität im IT-Bereich tatsächlich und potentiell jede einzelne IT-Fachkraft trifft, führt dazu, daß die berufliche Identität zum einzigen kontinuierlichen Selbstbezug in der Arbeit wird. Wir wollen im folgenden zunächst zwei Idealtypen der beruflichen Identität von IT-Fachkräften vorstellen: den »Hacker« und den »rationalen Techniker«.

---

[75] Eine Ausnahme von der Regel wäre tendenziell der Fall, in dem sich die berufliche Ausbildung auf einen Beruf beziehen würde, der auf von vornherein festgelegten, unabänderlichen Handlungssequenzen beruhen würde. Da es einen solchen Beruf nicht gibt und selbst hochstrukturierte Fließbandarbeit diesem Handlungsmodell nicht entspricht, bleibt dies ein theoretischer Fall. Trotzdem muß sich berufliche Ausbildung auf die berufliche Handlungswirklichkeit beziehen. Sie kann dies aber nur in vermittelter Form, das heißt, mittels plausibler heuristischer Modelle, deren Relevanz sie in der Praxis überprüfen muß.

5.3.1 Der »Hacker«

Das äußere Erscheinungsbild des Hackers ist von Weizenbaum eindringlich und gewissermaßen paradigmenbildend beschrieben worden. Wir zitieren dieses Erscheinungsbild hier ausführlich, um es anschließend auf seine Relevanz bezüglich der beruflichen Identität von IT-Fachkräften zu untersuchen:

"Überall, wo man Rechenzentren eingerichtet hat, d.h. an zahllosen Stellen in den USA wie in fast allen Industrieländern der Welt, kann man aufgeweckte junge Männer mit zerzaustem Haar beobachten, die oft mit tiefgesunkenen, brennenden Augen vor dem Bedienungspult sitzen; ihre Arme sind angewinkelt, und sie warten nur darauf, daß ihre Finger - zum Losschlagen bereit - auf die Knöpfe und Tasten zuschießen können, auf die sie genauso gebannt starren wie ein Spieler auf die rollenden Würfel. Nicht ganz so erstarrt sitzen sie oft an Tischen, die mit Computerausdrucken übersät sind, und brüten darüber wie Gelehrte, die von kabbalistischen Schriften besessen sind. Sie arbeiten bis zum Umfallen, zwanzig, dreißig Stunden am Stück. Wenn möglich, lassen sie sich ihr Essen bringen: Kaffee, Cola und belegte Brötchen. Wenn es sich einrichten läßt, schlafen sie sogar auf einer Liege neben dem Computer. Aber höchstens ein paar Stunden - dann geht es zurück zum Pult oder zum Drucker. Ihre verknautschten Anzüge, ihre ungewaschenen und unrasierten Gesichter und ihr ungekämmtes Haar bezeugen, wie sehr sie ihren Körper vernachlässigen und die Welt um sich herum vergessen. Zumindest solange sie derart gefangen sind, existieren sie nur durch und für den Computer. Das sind Computerfestischisten, zwanghafte Programmierer. Sie sind ein internationales Phänomen" [Weizenbaum 1977, S. 160f].

Wenn wir uns der beruflichen Identität von IT-Fachkräften annähern, werden wir sehr bald mit diesem klischeehaften Bild des Hackers konfrontiert. Dieses wird entweder als Mythos des genialen "garage programmers" [Rout 1992, S. 30] verklärt oder als Karikatur der eigenen beruflichen Identität angesehen. "Even the humour of the software industry supports this culture; the jokes about »real programmers« have more than a tinge of envy about them." [ebd.]. Da wir nur in den seltensten Fällen eine IT-Fachkraft treffen werden, die über eine berufliche Identität eines Hackers verfügt, könnte vorschnell behauptet werden, daß das Bild des Hackers oder »zwanghaften Programmierers« keine orientierende Funktion bei der beruflichen Identitätsbildung spielen würde. Wir behaupten dagegen, daß sich im ambivalenten Bild des Hackers die ambivalente Identitätsbewältigung der beruflichen Situation eines großen Teils der IT-Fachkräfte widerspiegelt. Dazu wollen wir kurz charakteristische Züge des Hackers skizzieren:

Der Hacker sieht sich nicht als technischer Ingenieur in einem Gestaltungsprozeß, sondern eher als **Künstler, der einen Programmcode**[76] gestaltet. "Good computer programmers have individual styles as all artist do. A programmer has as many choices in designing a word processor or game as a painter has in painting a still life or a writer has in writing a love story. It is making these choices that the programmer develops his own style and rhythm and becomes an artist" [Zaron, zit.n. Rout 1992, S. 30]. Dieses Künstlersein bedeutet einerseits, eine **hohe Individualität und Autonomie** für sich zu beanspruchen, andererseits erschwert es aber auch die Zusammenarbeit mit anderen Programmierern. Die Benutzung wiederverwendbarer Programm-Module erscheint für einen »Programmierkünstler« schon fast als Unzumutbarkeit. So sagt z.B. ein führender Programmierer von »Apple«: "Reusing other people's code would prove that I don't care about my work. I would no more reuse code than Hemingway would have reused other people's paragraphs" [zit.n. Rout 1992, S. 31]. Das Verstehen von Softwaresystemen, die der Programmierer nicht selbst geschrieben hat, wird als unmöglich angesehen, entsprechend redundant erscheinen dann oftmals auch die den Entwicklungsprozeß begleitenden Dokumentationen. Diese merkwürdige Ignoranz gegenüber fremden Programmierarbeiten findet sich schon in der Ausbildung wieder: "Computer science education has prepared developers with a background that emphasises fresh creativity almost exclusively. Students learn to work alone and to develop programs from scratch. They are rarely asked to understand software systems they have not written" [Shaw zit.n. Rout 1992, S. 31]. **Der Preis der Individualität und Autonomie ist die partielle Kooperationsunfähigkeit des einzelnen Programmierers.** Diese partielle Kooperationsunfähigkeit wird im Bild des Hackers in übertriebener Form hypostasiert und somit zurückgenommen: Der **Autismus des Hackers**, der mit seinem Computer verschmilzt und so aus der realen sozialen und natürlichen Welt flüchtet, wird nicht nur als ein zu hoher Preis für Individualität und Autonomie angesehen, sondern er stößt auch bald an innerbetriebliche Grenzen. "Er (der zwanghafte Programmierer) wird geschildert als ein brillanter Techniker, der zwar jedes Detail des Computers und des Systems kennt, dessen Arbeitsweise jedoch merkwürdig irrational atheoretisch ist, ohne klar definiertes langfristiges Ziel und ohne Plan zu dessen Verwirklichung, aber auch ohne Interesse an dem in das Programm zu übersetzenden Inhalt" [Beland 1988, S. 55] Der Umgang mit der partiellen Kooperationsunfähigkeit besteht freilich nicht nur in einem distanzierenden Humor, der die autistischen Auswüchse des Hackerseins thema-

[76] Diese Sichtweise teilt auch der Informatik-Professor Molzberger in seinem Aufsatz »Und Programmieren ist doch eine Kunst« [vgl. Molzberger 1983].

tisiert, sondern auch in der Bildung eines Fachjargons[77], der Kommunikation »entlang des Programmcodes« ermöglicht.

Zusammengefaßt können wir festhalten: Auch wenn IT-Fachkräfte fast nie die berufliche Identität eines Hackers besitzen, so dienen ihnen doch Bild und Mythos des Hackers als Referenz, um Individualität und Autonomie in ihrer beruflichen Identität zu verankern und berufsgruppenspezifische Sprachspiele zu initieren, die die partielle Kooperationsunfähigkeit, die aus der Fixierung auf den Programmcode entspringt, scheinbar überbrücken.

5.3.2 Der »rationale Techniker«

Der Techniker unterscheidet sich vom Hacker grundlegend: Anstelle des Künstlersein tritt die Anwendung von prinzipiell allen zugänglichen technischen Methoden. Das irrationale Element, das den Hacker kennzeichnet, weicht der Rationalität technischer Verfügbarkeit. Der Fachjargon wird durch eine Fachsprache abgelöst, die zumindest den Anspruch hat, jedes technische Programmierproblem anderen Technikern zugänglich zu machen. Der Techniker bejaht und fördert technische **Standardisierungsprozesse** sowohl bezüglich der Arbeitsmethoden als auch der Arbeitsprodukte. Er bezieht seine berufliche Identität aus der zertifizierten Zugehörigkeit zu einer Profession, die sich um "die Objektivierung beruflichen Wissens, die Loslösung von dem besonderen Talent, Charisma oder außergewöhnlichen, in früheren Zeiten häufig mystisch verklärten Fähigkeiten einzelner Individuen" bemüht [Hartmann 1993, S. 394]. Weizenbaum stellt den »Berufsprogrammierer« als positives idealtypisches Bild dem »zwanghaften Programmier« gegenüber:

"Der normale Programmierer diskutiert in der Regel sowohl sein inhaltliches als auch sein technisches Problem mit anderen (...). Seine eigentliche Rechenzeit am Computer ist relativ kurz. Es kann sogar vorkommen, daß er die Bedienung anderen überläßt. Sein Programm entwickelt er langsam und systematisch. Funktioniert etwas nicht, so überlegt er sorgfältig, wo die Ursachen dafür liegen könnten, und entwickelt Testprogramme, um die Fehler zu finden. Hier kann er den Probelauf wiederum von anderen fahren lassen. Während er auf die Ergebnisse wartet, kann er sich durchaus anderen Bereichen seiner Arbeit widmen." [Weizenbaum 1977, S. 162].

Die so beschriebenen Berufsprogrammierer haben eine normale[78] berufliche Identität in dem Sinne, daß sie über eine **affektive Distanz** und **soziale Kompe-**

---

[77] vgl. zu dem psychologisch interessanten assoziativen Gehalt dieses Fachjargons: [Huebner u.a. 1988].

tenzen verfügen, die sie befähigen, Technik als rational herstellbares Mittel für einen bestimmten Zweck zu gestalten, ohne die Grenzen des Selbst, ihre Ich-Identität, zu gefährden. Die Entwicklung dieser beruflichen Identität kann aus der **technischen Verwissenschaftlichung** und den mit ihr einhergehenden **Standardisierungsprozessen** sowohl der Software-Entwicklung als auch der Softwareprodukte abgeleitet werden, wie sie nicht zuletzt in der Ausbildung vermittelt werden. Es ist der notwendige Realitätsbezug der Identität, der zur Ausbildung und Durchsetzung dieser Identität des rationalen Technikers geführt hat.

Ist also die berufliche Identität des Hackers für die normale IT-Fachkraft obsolet geworden? Wir glauben dies kaum und zwar aus folgenden Gründen:

- Die Standardisierungsbemühungen sind trotz großer Erfolge weder bezüglich der Software-Entwicklung noch bezüglich der Softwareprodukte so weit vorangeschritten, daß sie den alleinigen Realitätsbezug der IT-Fachkräfte bilden können. Das Tüfteln am Programmcode ohne Rückgriff auf standardisierte Methoden ist nach wie vor notwendig. Selbst "die Case-Euphorie der letzten Jahre (ist) schlagartig verflogen, hat einer Ernüchterung und z.T. sogar Enttäuschung Platz gemacht" [Hartmann 1993, S. 413]. Das »Hackersein« findet somit immer noch **subjektive Legitimation des »Hackenmüssens«**, das »Künstlertum« noch immer seine betrieblichen Nischen [vgl. ebd., S. 407].
- Die Standardisierungsbemühungen werden zwar allgemein von den IT-Fachkräften begrüßt, aber letztendlich ambivalent bewertet. Dazu eine typische Äußerung eines Informatikers: "Ich bin für mehr Software-Engineering, obwohl das jetzt eine schöne Zeit ist. Man hat Freiheiten, die ein Bauingenieur nicht hat, weil die feste Regeln haben, wie was zu machen ist. Dagegen sagt der Informatiker, wenn er ein Problem hat, heute häufig noch: Ach, wie kann ich das denn mal machen." [zit.n. Hartmann 1993, S. 402]. Die Individualität und Autonomie, die ein grundlegendes Charaktermerkmal der beruflichen Identität des Hackers ist, wird durch Standardisierungsprozesse

---

[78] Von »normal« sprechen wir hier in Entgegensetzung zur unnormalen Identität des Hackers, die psychische Störungen beinhaltet, die unter dem Begriff des »borderline-Syndroms« populär geworden sind. Das heißt: Normalität bezeichnet in diesem Falle ein gesellschaftlich definiertes psychologisches Konstrukt, das sich zuallererst bezüglich des Vorhandenseins sozialer Kompetenzen zeigt. Daß auch der Typus des Hackers über positive Eigenschaften wie Kreativität, hohe Motivation und Flexibilität verfügt, ist unbestritten. Um diese Eigenschaften aber auch für die kooperative Gestaltung von Arbeit zu nutzen, müssen sie mit einer neuen beruflichen Identität verbunden werden.

gefährdet, die zwar rational positiv bewertet, aber affektiv oftmals als Einschränkungen abgelehnt werden. Mit der subjektiven Legitimation des »Hackenmüssens« korrespondiert nicht selten ein **unterschwelliges Bedürfnis des »Hackenwollens«**.

- Johnson hat aufgezeigt, daß die von Weizenbaum den Hackern zugeschriebenen Verhaltensweisen unter bestimmten Bedingungen zeitweilig "beim Tun der »Profis« eine Rolle spielen, und zwar unter Nichtbeinträchtigung ihrer Arbeitsdisziplin und technischen Kompetenz" [Johnson 1988, S. 152]. Die äußeren Bedingungen dafür hat er als Organisationskrise analysiert: Ausgangspunkt bildet eine **Organisationskrise**, die die Bewältigung von Umweltproblemen (konkurrenzbedingter erhöhter Druck zur Rationalisierung oder Produktentwicklung) auf die arbeitenden Individuen abwälzt, indem sie Überstunden und höhere Arbeitsintensität vom einzelnen einfordert und über die **Mobilisierung von distanzloser Arbeitsidentifikation** durchsetzt. Den im Zuge solcher Organisationskrisen beobachtbaren, ganze Entwicklungsabteilungen betreffenden Rückbezug der beruflichen Identität des rationalen Technikers auf die berufliche Identität des Hackers bezeichnet Johnson als **"Gruppenregression"** [ebd.]. Wir wollen hier nicht auf die z.T. problematischen theoretischen Implikationen von Johnsons Analyse eingehen, sondern das Vorhandensein des Phänomens selbst betonen: Die berufliche Identität des Technikers kann sich - und tut dies tatsächlich - in bestimmten Situationen in die berufliche Identität des Hackers transformieren und eine entsprechende Realitätssicht generieren.

Zusammenfassend können wir sagen, daß die Veränderung der beruflichen Realität (Standardisierungsprozesse) der IT-Fachkräfte und eine darauf zurückgehende, organisierte IT-Ausbildung zur Transformation der Identität dieser Berufsgruppe geführt hat. Die berufliche Identität des Hackers ist durch die berufliche Identität des rationalen Technikers verdrängt worden. Diese Verdrängung hat nicht bruchlos und nicht vollständig stattgefunden. Vielmehr finden auch heute noch zahlreiche Identitätskonflikte von IT-Fachkräften im Beruf und in der Ausbildung statt, die zwischen den zwei Polen Hacker und rationaler Techniker verortet werden können.

Gleichzeitig, so unsere Behauptung, strukturiert sich die betriebliche Rolle der IT-Fachkräfte aufgrund ökonomischer und unternehmensorganisatorischer Strategien gegenwärtig wiederum um und schafft somit eine neue Realität, die eine weitere Transformation der beruflichen Identität notwendig macht, um die berufliche Handlungskompetenz zu erhalten. Eine Didaktik, die auf die Entwicklung der beruflichen Handlungskompetenz ausgerichtet ist, muß diesen Trans-

formationsprozeß konstruktiv und aktiv begleiten, sonst läuft sie Gefahr, daß ihre formalen Inhalte an »klassischen« Identitäten wirkungslos abprallen.

5.4 Ganzheitliche Arbeitsgestaltungskompetenz und die notwendige Transformation beruflicher Identität

Weder die Identität des Hackers noch die Identität des rationalen Technikers können als Hintergrund der Ganzheitlichen Arbeitsgestaltungskompetenz dienen. Dies wird deutlich, wenn wir die am Anfang des Kapitels erläuterten drei Aspekte von Identität betrachten und sie auf die Identität des rationalen Technikers beziehen[79]:

- Die **subjektive Folie des Wahrnehmens und Erkennens** ist allein auf technische Probleme ausgerichtet. Sowohl der Software-Entwicklungsprozeß als auch das Softwareprodukt werden als rationale technische Problemstellungen angesehen. Der Charakter des sozialen Prozesses der Software-Entwicklung und das Ziel des technischen Produkts, Arbeit zu gestalten, werden als Störungen wahrgenommen, die es gilt zu minimieren. Dies geschieht dadurch, daß man diese Dimensionen ignoriert, beziehungsweise in den Verantwortungsbereich anderer Berufe verweist, oder dadurch, daß man sie möglichst schnell und eindeutig abhandelt. "Am besten ist, man hat schon beim zweiten Treffen mit den Anwendern das verbindliche Pflichtenheft erstellt", so die Aussage eines technisch orientierten IT-Ausbilders.
- Für die Entwickler-Benutzer-Kommunikation steht dem »rationalen Techniker« weder ein adäquates »Sprachspiel« zur Verfügung, noch ein dezentriertes Vorstellungssystem, das ihm ein empathisches Hineinversetzen in den Benutzer ermöglichen würde. Die berufliche Identität als rationaler Techniker ermöglicht nur die Kommunikation zwischen Entwicklern über den technisch richtigen Weg. Soziale Kompetenzen, die zu einer adäquaten Interaktion mit dem Benutzer beitragen, werden nicht durch die berufliche Identität gestützt, das heißt, für das berufliche Handeln als wertvoll erachtet. Folgerichtig erscheint die **Benutzer-Entwickler-Interaktion im Bewußtsein rationaler Techniker als Schnittstellenproblem**, das durch eine möglichst eindeutige Informationsübergabe zu bewältigen ist, und nicht als Interaktionssystem, in dem die eigentliche Software-Entwicklung stattfindet.

---

[79] Den Hacker wollen wir hier nicht betrachten, da seine Defizite allgemein anerkannt sind. Gerade die Etablierung des Unterrichtsfaches Software Engineering hat sich in ihrer Hauptströmung der Ausbildung des rationalen Technikers verpflichtet und dezidiert vom Mythos des kreativen Hackers abgesetzt.

- Wenn sich die betriebliche Realität dahingehend ändert, daß das Gestalten von Arbeitssystemen zum festen Bestandteil der Rollenerwartungen wird, erleidet der rationale Techniker zwangsläufig einen **Realitätsverlust**, der sich als subjektive Identitätskrise äußern kann. "»Problematisch« sind nicht diejenigen, die unter einer beruflichen Identitätskrise leiden, sondern die, die das nicht tun." [Siebert 1985, S. 14]. Dieser im Zusammenhang mit der beruflichen Identität von in der Erwachsenenbildung Tätigen geäußerte Satz Sieberts faßt Identitätskrisen als beginnende Reflexion von Realität und eigener Identität auf. Er steht im Gegensatz zu Formen der Realitätsleugnung und des resignativen Rückzugs. In diesem Sinne kann die Initiierung und Aufarbeitung von Identitätskrisen durchaus zum didaktischen Bestandteil der Ausbildung werden. Dazu ist es allerdings notwendig, daß die Realität auch Einzug in die Ausbildung erhält, was äußerst schwierig ist.

Da es zwar möglich ist, die Defizite vorhandener beruflicher Identitäten bezüglich einer sich ändernden Realität mit neuen mentalen Anforderungen und neuen Interaktionsbeziehungen zu benennen, es aber gleichzeitig nicht möglich ist, eine neue Identität, die sich erst in einer komplexen Auseinandersetzung mit diesen Dimensionen bildet, kohärent zu beschreiben, können wir hier nur eine grobe Skizze adäquater beruflicher Identität liefern. In früheren Veröffentlichungen [Baukrowitz 1992] haben wir die, den Erfordernissen der Ganzheitlichen Arbeitsgestaltungskompetenz gerecht werdende, neue berufliche Identität analog zur neuen betrieblichen Rolle als **»Prozeßmoderator«** bezeichnet. Diese Bezeichnung haben wir bewußt als Abgrenzung von technikzentrierten Begriffen gwählt. Sie kennzeichnet u.E. auch den Kernbereich zukünftiger IT-Arbeit: Die IT-Fachkraft gestaltet Arbeit im sozialen Prozeß der Software-Entwicklung. Dabei bezieht sie technische Möglichkeiten auf im sozialen Prozeß generierte Arbeitsgestaltungsalternativen und macht beide den Prozeßteilnehmern kommunikativ (über sprachliche, graphische Medien oder technische Medien: Prototypen) zugänglich, die gemeinsam wiederum neue Arbeitsgestaltungsalternativen generieren usw. Da der Begriff des Moderators allerdings nicht nur alltagssprachlich, sondern auch im Duden mit der Tätigkeit des Fernseh- und Rundfunkmoderators gleichgesetzt wird, birgt der Begriff des Prozeßmoderators mehr Anlaß zu Mißverständnissen als zur Klärung einer neuen beruflichen Identität. Wir werden daher diesen Begriff nur als Chiffre einer neuen beruflichen Identität einsetzen.

5.4.1 Der »Prozeßmoderator« - Von der technischen Verantwortung zur Prozeßverantwortlichkeit

Mit dem Konzept der qualitativen Bestimmung von Verantwortung als Funktion sozialer Systeme hat Kaufmann ein Modell der betrieblichen Rollenzuschreibung

entwickelt, auf das wir uns beziehen können, wenn wir die Identität des Prozeßmoderators von der Identität des rationalen Technikers systematisch abgrenzen wollen. Erst diese systematische Abgrenzung macht deutlich, daß der Prozeßmoderator kein um ein paar soziale Kompetenzen ergänzter Techniker ist, sondern eine qualitativ neue Form der beruflichen Identität darstellt. Bevor wir eine Matrix zur qualitativen Bestimmung des Prozeßmoderators vorstellen werden, erscheint es uns notwendig, in kurzer Form auf die **organisatorische Ausprägung von Verantwortungszuschreibungen im Arbeitsprozeß** einzugehen. Wir unterscheiden prinzipiell drei Formen von Verantwortungszuschreibungen:

Zuverlässigkeit: Einfache Zuverlässigkeit bedeutet in der von uns vorgeschlagenen Definition das zuverlässige Ausführen einer einzelnen fachlichen Tätigkeit. Zuverlässigkeit kann so gleichermaßen das korrekte Ausfüllen eines Formulars wie die korrekte Montage eines Maschinenteils bezeichnen.

Verantwortung: "Verantwortung meint die Zuschreibung an eine Position oder Rolle, deren Aufgabenspektrum durch einen erheblichen Handlungsspielraum und entsprechende selbständige Entscheidungszumutungen sowie hohes Folgerisiko gekennzeichnet ist" [Kaufmann zit.n. Corsten/ Lempert 1992, S. 64]. In der Verantwortung ist Zuverlässigkeit aufgehoben.

Verantwortlichkeit: "Verantwortlichkeit meint die Zuschreibung an Personen bzw. Positionsinhaber als Fähigkeit, bestimmten Verantwortungen, insbesondere jedoch der Kombination mehrerer und potentiell konfligierender Verantwortungen gerecht zu werden" [ebd.]. Verantwortungen sind in der Verantwortlichkeit aufgehoben und stärker auf die Person als auf die Rolle bezogen. Das heißt kongruent für die Ausbildung der beruflichen Identität: Berufliche Identität ist näher an der Ich-Identität als an einer betrieblich diffusen Rollenidentität angesiedelt.

Wenn wir diese Formen der Verantwortungszuschreibungen mit von uns eingeführten Begrifflichkeiten verbinden, erhalten wir folgende Matrix:

	Sichtweise der Aufgaben	**spezifische Handlungskompetenz**	**Identitätsleistung**
Zuverlässigkeit	algorithmische Sichtweise	sequentielle Handlungskompetenz (Ausführen)	sekundäre Identitätsleistung aufgrund einer tätigkeitsorientierten beruflichen Identität (Arbeitstugenden)
Verantwortung	fachspezifische Sichtweise	vollständige Handlungskompetenz (Planen & Ausführen)	rollenorientierte Identitätsleistung aufgrund einer fachspezifischen beruflichen Identität
Verantwortlichkeit	multiperspektivische Sichtweise	Ganzheitliche Handlungskompetenz (übergreifendes Entscheiden & Planen & Ausführen)	Ich-Identitätsorientierte und rollenorientierte Identitätsleistung aufgrund einer arbeitsprozeßbezogenen reflektierten beruflichen Identität

Weitere, in dieser Matrix nicht aufgeführte, aber trotzdem erwähnenswerte Dimensionen sind die Gruppenorientierung (Arbeiter/ Facharbeiter/ Profession) und der nicht zuletzt auch aus der Höhe der Entlohnung bestimmte soziale Status (niedrig/ mittel bis hoch/ sehr hoch), der mit den unterschiedlichen Verantwortungszuschreibungen jeweils verbunden ist.

Bei näherer Betrachtung dieser Matrix und der Rekapitulation des Software-Entwicklungsprozesses als sozialem Prozeß, in dem primär Arbeit und nur sekundär Technik gestaltet wird, sehen wir, daß die betriebliche Organisation dieses Prozesses auf **Verantwortungszuschreibungen rekurrieren muß, die der Form der Verantwortlichkeit entsprechen**. Eine ausführlichere Definition des Begriffs der Verantwortlichkeit verdeutlicht dies. Folgende Beschreibung von Verantwortlichkeit kommt durch die Zusammenziehung mehrerer Textpassagen von Kaufmann zustande und liest sich fast wie eine Beschreibung des »Prozeßmoderators« im Systementwicklungsprozeß:

"Wir können (...) Verantwortlichkeit (...) als Fähigkeit zum Ausgleich unterschiedlicher Rollenverantwortungen interpretieren. (...) Verantwortlichkeit ist gerade dort gefragt, wo die herkömmlichen Mittel der Definition und Kontrolle von Pflichten versagen. (...) Die soziale Funktion der Zuschreibung von Verantwortung und Verantwortlichkeit besteht (...) in der Mobilisierung von Selbstverpflichtung im Sinne außergewöhnlicher, nicht programmierbarer Handlungsbereitschaft für spezifische Zwecke sozialer Systeme. (...) Die Zuweisung von Verantwortung ist (...) ein Korrelat von Vertrauen in die angemessene Erfüllung einer

hinsichtlich ihres Ergebnisses unabsehbaren Aufgabe" [Kaufmann zit.n. Corsten/ Lempert 1992, S. 108].

Der **»Prozeßmoderator«** übernimmt die **Verantwortlichkeit** für die Software-Entwicklung als Arbeitsgestaltungsprozeß, während der rationale **Techniker** nur die **Verantwortung** für die Lösung spezifischer technischer Probleme übernimmt. Die Übernahme der Verantwortlichkeit heißt nicht, daß er jede Aufgabe des Prozesses sich zu eigen macht, also alle Verantwortungen übernimmt, sondern daß er unterschiedliche Verantwortungen aufeinander bezieht und selbst Verantwortungen definiert. Die zur Übernahme dieser Funktion notwendige Identitätsleistung bezeichnen wir als **arbeitsprozeßbezogene berufliche Identität**. Diese Identität hat starke Bezüge zur Ich-Identität und ist nur in geringem Maße rollenbezogen. Dies liegt u.a. daran, daß es **keine kohärente Rolle des Prozeßmoderators** gibt, also keinen festgelegten Katalog von Verhaltenserwartungen, sondern nur ein Bündel unterschiedlichster Rollen, die vom Individuum personal integriert werden müssen. Insbesondere wird damit auch deutlich, welche hervorgehobene Funktion diese berufliche Identität bezüglich der Teilkompetenzen der Ganzheitlichen Arbeitsgestaltungskompetenz einnimmt:

Die Technikgestaltungskompetenz, die Prozeßgestaltungskompetenz und die Kooperationskompetenz spiegeln unterschiedliche Rollenbezüge wider, die nur durch **integrierende Identitätsleistungen** des Subjekts zu einer kohärenten Ganzheitlichen Arbeitsgestaltungskompetenz werden **können.** Die der Systemlogik des Arbeitsprozesses entspringende Ganzheitlichkeit des Arbeitsgestaltungsprozesses Software-Entwicklung ist nicht mehr durch eine auf festen Verhaltensschemata beruhende Rollenidentität auf Seiten des handelnden Subjekts realisierbar, sondern nur durch eine arbeitsprozeßbezogene berufliche Identität. Diese hat daher über weite Strecken den Charakter eines **reflexiven Subjektivismus.**[80]

Der von uns gemachte Vorschlag, IT-Fachkräfte für die Funktion des Prozeßmoderators ganzheitlich zu qualifizieren, hat auf Seiten der Ausbilder zwei vehemente Einwände hervorgerufen, auf die wir an dieser Stelle eingehen müssen:

1. Die Ausrichtung der Qualifizierung für die Funktion des Prozeßmoderators entspricht einer Ausbildung von Projektleitern. Projektleiterstellen werden aber zunehmend von Informatikern mit Hochschulausbildung bzw. mit Wirt-

---

[80] Der Systemtheoretiker Schimank behauptet, daß diese Identitätsform generell funktional differenzierten Gesellschaften entsprechen würde [vgl. Schimank 1985], weil "die Koordination und Integration der verschiedenen Rollendarstellungen nicht länger sozial reguliert (werden kann), sondern (...) auf die Person abgewälzt werden (muß)" [Schimank 1981, S 23].

schaftsinformatikern besetzt. Informationstechnische Ausbildungsberufe können nicht mit den Qualifikationsprofilen universitärer Ausbildung bezüglich dieser Positionen konkurrieren. Vielmehr liegt ihre Stärke in einer soliden, praktisch orientierten technischen Ausbildung. Die Qualifikation zur Übernahme technischer Verantwortung ist angesichts der permanenten technischen Innovationen schon schwer genug.

2. Die Übernahme von Verantwortlichkeit für die Software-Entwicklung als Arbeitsgestaltungsprozeß befreit die einzelne IT-Fachkraft nicht von ihrer Verantwortung für die Lösung spezifischer technischer Probleme. Das heißt, der qualitative Sprung vom rationalen Techniker zum Prozeßmoderator schlägt sich nicht nur in der qualitativen Veränderung der Ausbildungsinhalte nieder, sondern führt auch zur quantitativen Steigerung der Lerninhalte. Da diese aber in der beruflichen Ausbildung nicht zu leisten ist, kann die Qualifizierung zum Prozeßmoderator nur auf Kosten der technischen Qualifizierung gehen. Dieses führt aber zu einem immanenten Widerspruch, da technische Qualifizierung auch eine Voraussetzung für die Übernahme der Verantwortlichkeit für die Software-Entwicklung ist.

Beide Einwände ergänzen sich: Für die Position des Prozeßmoderators qualifizieren andere Ausbildungsrichtungen und selbst wenn wir es wollten, so könnten wir es gar nicht. Unsere Gegeneinwände lauten:

zu 1. Die Funktion des Prozeßmoderators beschränkt sich keineswegs auf die Position des Projektleiters in großen Projekten. Selbst die permanente Anpassung einer Standardsoftware innerhalb einer Fachabteilung, die einer einzigen IT-Fachkraft übertragen wird, entspricht dem Paradigma der Arbeitsgestaltung, ist also in ihren unterschiedlichen Bezügen nur durch die Übernahme von Verantwortlichkeit, die weit über die technische Verantwortung hinausreicht, bewältigbar. Zudem werden auch in jedem größeren Projekt der einzelnen IT-Fachkraft Verantwortungen übertragen, die sie nur auf der Folie der inneren Übernahme von überschreitender Verantwortlichkeit (z.B. Bildschirmgestaltung) und nicht durch den Rückzug auf eine rein technische Orientierung wahrnehmen kann. Die Komplexität des Software-Entwicklungsprozesses erzwingt es gewissermaßen, daß der Projektleiter seine Gesamtverantwortlichkeit nicht allein durch das Delegieren von spezifischen Verantwortungen bewältigen kann, sondern immer auch auf die Delegation von Verantwortlichkeit angewiesen ist. Auch der einzelne, arbeitsteilig arbeitende Programmierer muß sich auf den Gesamtprozeß der Arbeitsgestaltung beziehen.

zu 2. Die Fähigkeit, die Funktion des Prozeßmoderators einzunehmen, ist in erster Linie eine Identitätsleistung. Identität wird aber nicht im klassischen Sinne

gelehrt, sondern durch spezifische Erfahrungen in einem spezifischen sozialen Kommunikationszusammenhang erworben. Die dazu notwendigen sozialen Kompetenzen werden ebenfalls nicht in Form zusätzlicher Unterrichtsinhalte, sondern durch eine konzeptionelle Neugestaltung des Unterrichts gefördert. Identitätsbildung und die Förderung von sozialen Kompetenzen liegen gewissermaßen quer zu den traditionellen Unterrichtsinhalten. Die grundlegenden Wissensinhalte, die mit der Funktion des Prozeßmoderators verbunden sind, haben wir als Erschließungswissen in dem Qualifizierungsbaustein »Aufgabenverständnis« und dem Rahmencurriculum »Moderne Software-Entwicklung« dargelegt.

Mit unserem zweiten Gegeneinwand haben wir faktisch die **Frage nach der Genese von Identität und ihrem Verhältnis zur Wissensdimension** der beruflichen Handlungskompetenz gestellt. Damit rückt wieder die von uns zu Anfang des Kapitels zitierte Definition von Identität in den Mittelpunkt unserer Betrachtung, die Identität weitgehend als System von Regeln und Interpretationsmustern faßt. Die Diskussion dieser Frage soll den Abschluß unserer Erörterung der Problematik beruflicher Identität bilden.

5.5 Identitätsbildung als Integration von Vorerfahrungen

Bei der Betrachtung der Wissensdimension der beruflichen Handlungskompetenz stellten wir fest, daß die handelnden Subjekte zwar in einem objektiven Handlungsspielraum agieren, ihre Handlungen aber nur in einem **subjektiv konstruierten Handlungsraum** entwerfen und einordnen können. Dieser subjektiv konstruierte Handlungsraum ist nun maßgeblich nicht nur durch die aktuelle Situation bestimmt, sondern dadurch, daß die Handelnden die aktuelle Situation schon beim Prozeß des Wahrnehmens auf subjektive Vorerfahrungen und subjektives Vorwissen beziehen müssen. Erfahrung beruht immer auf Vorwissen, das als verallgemeinerte Vorerfahrung bezeichnet werden kann. Hier kann ein ewiger Rekurs einsetzen, da jeder Vorerfahrung wiederum eine Vorerfahrung zugrunde liegt.

Dies entspricht einer behavioristischen Sichtweise, die - außer der Dimension einer positiven bzw. negativen Besetzung (Belohnung/ Strafe) einzelner Vorerfahrungen - kein weiteres strukturierendes Merkmal der Menge der subjektiven Vorerfahrungen für gültig ansieht. Komplexe berufliche Handlungspläne können aber kaum aus isolierten Vorerfahrungen gewonnen werden. Es stellt sich sogar die Frage, wie Subjekte überhaupt systematisch zu neuen Handlungsplänen kommen können, wenn sie alte Erfahrungen nur rekapitulieren. Selbst die behavioristische Erweiterung, daß Subjekte mittels des Modellernens auch Erfahrungen anderer Individuen als eigene Vorerfahrungen aufbauen können, löst

dieses Problem nicht, sondern erweitert es nur. Der behavioristischen Lern- und Verhaltenstheorie deshalb generell ihre Gültigkeit abzusprechen, hieße aber, diese Theorie an einem Anspruch zu messen, den sie selbst weit von sich weist. Handlungen, die auf einer komplexen Integration unterschiedlicher Vorerfahrungen beruhen, entziehen sich der wissenschaftlichen Analyse, da nur das Resultat, aber nicht der Prozeß dieser kognitiven Integration beobachtbar ist.

Das heißt für uns: Probleme, die wir als Probleme der beruflichen Identität beschreiben, wie z.B. die Art und Weise, wie Probleme des Arbeitsprozesses als berufliche Aufgaben erkannt und bearbeitet werden, erscheinen unter der Perspektive des Behaviorismus als eine spezifische Konditionierung des beruflich Handelnden. Können sie nicht mehr als solche erfaßt werden, entziehen sie sich prinzipiell unserer Erkenntnis. Der Begriff der beruflichen Identität ist demnach entweder wissenschaftlich als »berufliche« Konditionierung zu beschreiben oder als spekulativer Begriff anzusehen. Für eine wissenschaftliche Betrachtung ist er somit redundant. Auch wenn wir diese behavioristische Perspektive im folgenden ablehnen und am Begriff der beruflichen Identität festhalten, sehen wir die **Bedeutung »beruflicher« Konditionierung für den Aufbau einer bestimmten beruflichen Identität** ein, ohne daß letztere allerdings auf erstere reduzierbar wäre.

Die Kognitionspsychologie Piagets und die Theorien der Handlungsregulation beschäftigen sich im Gegensatz zum Behaviorismus mit den Integrationsleistungen eines kognitiven, handelnden Subjekts. Verschiedene Vorerfahrungen werden durch Regeln geordnet und kausal miteinander verbunden. Diese ordnenden und kausalverknüpfenden Regeln werden wiederum in immer komplexeren Systemen integriert und gewinnen eine abstrakte Gestalt. Abstrakte Ordnungsregeln finden wir paradigmatisch in der Mengenlehre, abstrakte Kausalitätsregeln in der formalen Logik und abstrakte Regeln des Handelns in den Prinzipien des hierarchisch-sequentiellen Handlungsentwurfs. Allen diesen Regeln ist gemeinsam, daß sie von einem autonomen Subjekt nicht allein aufgrund von Erfahrungen aufgebaut werden, wie dies bei konditionierten Verhaltens- und Denkweisen der Fall ist, sondern daß sie eine eigenständige Konstruktion des Individuums darstellen. Als eigenständige Konstruktionen folgen diese Regeln auch einer eigenständigen Entwicklung, die Erfahrungen zwar als Anstoß für Veränderungen benötigt, diese Veränderungen aber nur durch das Zurückgreifen auf schon vorhandene, eigene Regeln umsetzen kann.[81]

---

[81] Das Begreifen der Wirklichkeit kann somit auch als Verknüpfung von Erfahrungen mit abstrakten Regeln beschrieben werden. Diese Verknüpfung bezeichnen wir als Begriff.

Dieses kybernetische Postulat der Selbstorganisation[82] bezieht sich nicht nur auf die Einordnung einer neuen Erfahrung durch das Subjekt, sondern weitergehend auch schon auf die Form der Wahrnehmung dieser Erfahrung. Ich sehe nur die Phänomene als handlungsrelevant an, die ich mit meinem Handlungshorizont verbinden kann. Das heißt, der Fokus der subjektiven Aufmerksamkeit wird durch das kognitive System des Subjekts bestimmt. Entwicklung ist, plastisch ausgedrückt, nur an den »Rändern« möglich. Zur Veranschaulichung: Ein Lehrer will den Schülern eine generell neue Sichtweise eines bestimmten Phänomens nahebringen, die ihm selbst vollkommen plausibel erscheint. Trotz aller didaktischer Raffinessen wird er scheitern, wenn er sie nicht schrittweise aus den bei den Schülern vorhandenen Sichtweisen ableitet. Die kybernetische Selbstorganisation der Handlungsorientierung, die als Prozeß von Wahrnehmung, Erfahrungshorizont, abstrakten Regeln und Begriffen beschrieben werden kann, genügt nun vollkommen, um das Phänomen des subjektiven Handlungsraums strukturierend zu erklären. Der Rückgriff auf den Begriff der beruflichen Identität erscheint auch hier als redundant.

Auch wenn wir weitestgehend die vorstehenden theoretischen Annahmen teilen, so glauben wir, doch zeigen zu können, daß sich der Begriff der beruflichen Identität nicht auf diese psychologischen Theoreme reduzieren läßt, sondern auch **in seiner Genese soziale Bezüge aufweist**, die über diese hinausgehen. Das Ausblenden des kommunikativen Handelns als Konstitivum des wahrnehmenden und erkennenden Subjekts finden wir sowohl in der Kognitionspsychologie Piagets als auch in der Handlungsregulationstheorie. Dieses Ausblenden führt aber zu theoretischen Verengungen, die u.E. keine angemessene Beschreibung der Grundlagen menschlicher Handlungsfähigkeit ermöglichen. Der nachstehende Vorschlag nimmt nun die Grundannahmen der genannten Erklärungsmodelle (vgl. Tabelle) auf, um sie mit sozialen Bezügen zu ergänzen.

Wir rekapitulieren: Die Summe der Vorerfahrungen bildet den **Handlungshorizont** des Individuums, dessen Wirkungsmächtigkeit sich auch in der umgangssprachlichen Rede, daß manche Dinge irgendjemandes Horizont übersteigen würden, wiederfindet. Dieser Handlungshorizont stellt allerdings nicht nur die Grundlage der selbstorganisierten Konstruktion des subjektiven Handlungsraums aus Erfahrungen und formalen Verknüpfungsregeln dar, sondern ist u.E.

---

[82] Ein berechtigter Vorwurf an die Selbstorganisationstheoretiker soll hier allerdings nicht verschwiegen werden: Da, "wo's um qualitative Veränderungen geht, (wird) von »Sprüngen« gesprochen, was nichts anderes heißt, als daß man im Dunkeln tappt, und zwar buchstäblich. (Ein ähnliches Problem kennen wir aus Piagets Entwicklungspsychologie: die »Sprünge« zu höheren kognitiven Strukturen können zwar festgestellt, letztlich aber nicht weiter erklärt werden.)" [Waldvogel 1990, S.51].

ebenfalls mit einem **subjektiven Sinnhorizont** verbunden, der die Summe der Vorerfahrungen mit einem **kohärenten Selbstbild im kommunikativen Handeln** verknüpft. Das heißt: Vorerfahrungen und Begriffe werden **nicht** im Gedächtnis abgelegt, um bei Bedarf aktuelle Erfahrungen zu strukturieren. Vielmehr werden Vorerfahrungen in einem prozeßhaften, subjektbezogenen **Sinnsystem** eingeordnet und ständig verändert. Vorerfahrungen ändern sich in diesem Sinnsystem des Subjekts, das auch als **biographisches Selbstbewußtsein** bezeichnet werden kann, nicht nur bezüglich ihrer emotionalen Besetzung, sondern auch inhaltlich. Sie entwickeln sich mit der Identitätsentwicklung, das heißt, immer auch im kommunikativen Bezug zu den konkreten Anderen. Die Annahme, daß das wahrnehmungsstrukturierende Vorwissen nicht allein durch einen unendlichen Rekurs auf immer weiter zurückliegende Vorerfahrungen rekonstruierbar ist, sondern daß dieses Vorwissen erst durch die **Integration** der einzelnen Erfahrungen durch das **identische Subjekt** zustande kommt, bildet den Ausgangspunkt der nachstehenden Überlegungen.

Wahrnehmungs- und erkenntnisstrukturierendes Vorwissen und Vorerfahrungen, die die Handlungsorientierung einzelner Subjekte bestimmen, in unterschiedlichen theoretischen Erklärungsmodellen

Wahrnehmungsstrukturierendes Vorwissen und Vorerfahrungen	Theoretisches Erklärungsmodell	Theoriebezüge
Subjekte verfügen über eine **Menge von Vorerfahrungen**, die durch einen bestimmten Reiz ausgelöst werden können, wobei diejenige Vorerfahrung handlungsleitend wird, die häufig zum **Erfolg** geführt hat, deren Begleitumstände (**Reize**) der aktuellen Situation am ehesten gleichen und die unter den bekannten Handlungsalternativen den größten Wert (**Kosten-Nutzen**-Aspekt der Belohnung) besitzt.	**Behavioristische Psychologie** und die sich auf sie beziehende **verhaltenstheoretische Soziologie**	Alle Verhaltensakte werden auf **Formen des Konditionierens**, das heißt Erfahrungen, zurückgeführt. Die Verwendung dieser Erfahrungen gehorcht dem **Rationalitätsprinzip**. Der Handelnde prüft die Adäquatheit (Reizähnlichkeit), die Erfolgswahrscheinlichkeit und den Nettonutzen (Kosten-Nutzen-Kalkül) der Handlungsalternativen.
Subjekte verfügen über **kognitive Handlungspläne**. Spezifische Handlungssituationen werden an diese Handlungspläne durch die Subjekte angeglichen (assimiliert). Falls diese Situationen sich als nicht unmittelbar angleichbare Realitäten erweisen, ist die Anpassung (Akkommodation) des Handlungsplans an die Situation erforderlich. Da tatsächlich das nach außen gerichtete Angleichen nicht von dem nach innen gerichteten Anpassen trennbar ist[83], spricht Piaget von "der nicht dissoziierbaren Interaktion von Assimilation und Akkommodation" [Piaget 1981, S. 209].	**Genetische Erkenntnistheorie Piagets** In Bezug auf handlungsrelevantes Denken: **Handlungsregulationstheorien**	Verhaltens- und Erkennensakte werden auf kognitive Strukturen rückbezogen, die zwar vorgängige Erfahrungen einbauen, sich aber nach dem eigenständigen Kriterium eines **inneren Gleichgewichts** (Äquilibration) strukturieren. Erfahrungswissen liegt demnach nicht als summarische Menge von unterschiedlich besetzten Erfahrungen, sondern als **Integrationsleistung eines kognitiven Subjekts** vor: Dabei steigert sich die Komplexität der Integration kontinuierlich vom Beherrschen adaptiver, motorischer Funktionen bis hin zum Beherrschen mathematisch-logischer Systeme.
Wahrnehmungsstrukturierendes Vorwissen ist nicht nur Teil eines autonomen Subjektes, sondern ist maßgeblich dadurch bestimmt, welche soziale **Rolle das Individuum** einnimmt **und** wie das Individuum sich selbst in dieser Rolle sieht (z.B. berufliche **Identität**). Diese rollenspezifischen Wahrnehmungsmuster haben einerseits für das Individuum entlastende und selbstwertstabilisierende Funktionen, andererseits können sie auch als sozialer Zwang (Erwartungsdruck) angesehen werden.	**Identitätstheorien**, die aus dem **symbolischen Interaktionismus** hervorgehen[84]	Die theoretischen Bezüge sind in erster Linie soziologischer Art und beziehen sich auf **gesellschaftliche Gruppen**. Die psychologische Genese und Stabilisierung dieser sozial bestimmten Wahrnehmungsmuster im einzelnen Individuum können nur zureichend erklärt werden, wenn behavioristische und kognitionspsychologische Ansätze integriert und um Aspekte der Motivations- und Emotionspsychologie ergänzt werden.

[83] Ein kurzer Hinweis: Die Unterscheidung der Richtungen (Innen/Außen) bei Aktivitäten des Erkennens wird in der Literatur unterschiedlich gehandhabt, was oftmals zu unnötigen Verwirrungen führt, da diese Richtungsangaben in erster Linie nur der Veranschaulichung dienen und sich je nach Perspektive ändern können.

[84] z.B. Habermas (siehe unsere Darstellung der sozialen Kompetenzen, insbesondere die Ausführungen zum Begriff der Rollendistanz).

Die Integration einzelner beruflicher Erfahrungen zu einem Vorwissen, das die beruflichen Wahrnehmungen und Handlungen von Akteuren bestimmt, kann u.E. unter mehreren Gesichtspunkten betrachtet werden:

- Welche Verhaltensakte werden durch die Ausbildung und die berufliche Arbeitssituation direkt durch Belohnung/ Bestrafung konditioniert? Wichtig ist uns hier, daß mit den Verhaltensakten auch bestimmte Formen der Wahrnehmung und des Denkens verbunden sind (**behavioristischer Aspekt der beruflichen Identität**).
- Welche logischen und empirischen Zusammenhänge zwischen den einzelnen Erfahrungen werden von den einzelnen Akteuren aktiv[85] hergestellt? Dies ist die Frage nach der sachlichen Dimension der Integration von Vorerfahrungen in Vorwissen durch das Individuum (**handlungsregulations- und kognitionstheoretischer Aspekt der beruflichen Identität**).
- Welche **Identitätsleistung** (der Rollenidentität und Ich-Identität) des beruflichen Akteurs fließt in die Integration einzelner beruflicher Erfahrungen ein? Dies beinhaltet die Frage, inwieweit das integrierte Vorwissen auf »Vorurteilen« beruht, die weder sachlich noch empirisch ableitbar sind, sondern selbst schon durch eine vorausgehende, spezifische Identität geprägt sind. Diese Identitätsbildung ist kommunikativem Handeln und somit auch didaktischem Handeln nur in herrschaftsfreier Kommunikation zugänglich, die Selbstreflexion zum Thema macht (**selbstreferentieller Aspekt der beruflichen Identität**).

Die Punkte bezeichnen grundlegende Faktoren der beruflichen Identitätsbildung und verweisen gleichzeitig auf didaktisch-methodische Zugänge zum Identitätslernen.

Damit wollen wir sowohl die Diskussion um die berufliche Identität als auch um die Wissensdimension und die sozialen Kompetenzen als Teilbereiche beruflicher Handlungskompetenz beenden. Die dabei hervorgetretenen zahlreichen unterrichtspraktischen Implikationen für eine angemessene didaktisch-methodische Umsetzung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« sind Gegenstand des folgenden Teils II unserer didaktischen Überlegungen.

---

[85] Zusammenhänge, die so offensichtlich erfahren werden, daß sie kaum aktiv hergestellt werden müssen, wollen wir hier nicht betrachten, da sie keine aktive Reflexion des Individuums voraussetzen.

6 Zwischenbetrachtung: Die Analyse beruflicher Handlungskompetenz und didaktisch-methodisches Handeln

Wir sind davon ausgegangen, daß erst eine exakte Analyse der subjektiven Grundlagen der beruflichen Handlungskompetenz zu einer angemessenen, didaktisch-methodischen Fundierung von Unterricht führt. Im ersten Teil dieser Arbeit haben wir diese Analyse theoriegeleitet durchgeführt. Bevor wir uns nun den didaktisch-methodischen Konsequenzen unserer theoretischen Bemühungen zuwenden wollen, erscheint uns eine kurze Zwischenbetrachtung angemessen. Dabei wollen wir uns auf zwei Fragen konzentrieren: Welche Ergebnisse können als Quintessenz unserer theoretischen Überlegungen betrachtet werden? Welchen praktischen Gewinn für die didaktisch-methodische Gestaltung von Unterricht implizieren diese Ergebnisse? Die Beantwortung der zweiten Frage wird ausführlich in den Kapiteln 4 bis 7 des nächsten Teils unserer Arbeit dargelegt. Daher werden wir sie hier nur kurz und in allgemeiner Form behandeln.

Unsere Hauptthesen des ersten Teils unserer Arbeit können in folgenden Sätzen zusammengefaßt werden:

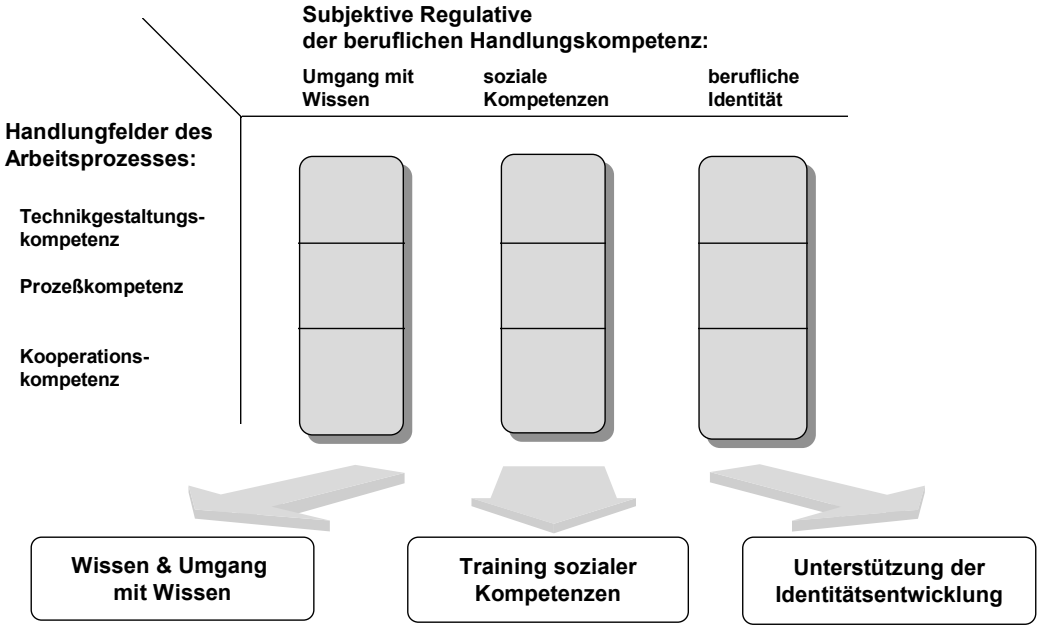
Eine Didaktik der Ganzheitlichen Arbeitsgestaltungskompetenz kann nicht dadurch fundiert werden, daß sie die vielfältigen Arbeitstätigkeiten des Arbeitsprozesses summarisch erfaßt und in einzelne Unterrichtseinheiten zerlegt. Die Gestaltung des Unterrichts kann sich nicht an der Simulation des Arbeitsprozesses orientieren. Auch die von uns genannten Teilkompetenzen der Ganzheitlichen Arbeitsgestaltungskompetenz, die Kernqualifikationen Technikgestaltungskompetenz, Kooperationskompetenz und Prozeßkompetenz beschreiben die Neuausrichtung der Qualifikation von IT-Fachkräften aus der Perspektive des Arbeitsprozesses. Ausgangspunkt der Entwicklung eines didaktisch-methodischen Konzepts bildet dagegen die folgende Frage: Welche subjektiven Faktoren benötigt das einzelne Individuum, um die Qualifikationsanforderungen des Arbeitsprozesses adäquat zu erfüllen?

Als solche Faktoren haben wir den subjektiven Umgang mit Wissen, das Verfügen über soziale Kompetenzen und die Entwicklung einer angemessenen beruflichen Identität analysiert. Diese Faktoren bilden zusammen die Grundlage der subjektiven beruflichen Handlungskompetenz. In Anlehnung an die psychologische Theorie der Handlungsregulation haben wir diese Faktoren auch präziser als Regulative des subjektiven Handelns bezeichnet. Gleichzeitig haben wir im zweiten Kapitel des ersten Teils unserer Ausführungen zur Didaktik darauf hingewiesen, daß diese subjektive Handlungskompetenz nicht losgelöst von dem konkreten Arbeitsprozeß betrachtet werden kann. Dies ist u.E. genau der Fehler

des Konzepts der Schlüsselqualifikationen, das beansprucht, eine allgemeine Problemlösefähigkeit und soziale Kompetenzen unabhängig von fachlichen Bezügen vermitteln zu können.

Das hieß für uns, daß wir die Bestandteile der subjektiven Handlungskompetenz jeweils auf die objektiven Anforderungen des Arbeitsprozesses beziehen mußten. Dieser Bezug, der jeweils den Abschluß der Kapitel darstellt, die den einzelnen Bestandteilen der Handlungskompetenz gewidmet sind, wird in folgender Matrix deutlich:

Didaktische Handlungsfelder



Mit Hilfe dieser Matrix haben wir nun drei theoretisch fundierte didaktische Handlungsfelder aus der Perspektive der beruflichen Handlungskompetenz beschrieben.

Die unterrichtspraktische Bedeutung dieser didaktischen Handlungsfelder wird in den Kapiteln 4 bis 7 des folgenden Teils unserer Arbeit ausführlich dargelegt. Dieser Teil unserer Arbeit ist primär didaktisch-methodisch ausgerichtet und hat keineswegs die Funktion, die zentralen Unterrichtsinhalte des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« zu beschreiben. Diese Aufgabe bleibt dem daran anschließenden Basisbaustein »Aufgabenverständnis« und dem Rahmencurriculum »Moderne Software-Entwicklung« vorbehalten.

Trotz dieser auch von uns vorgenommenen Trennung[86] zwischen curricularen, am Arbeitsprozeß orientierten Unterrichtsinhalten und den Vorschlägen der didaktisch-methodischen Gestaltung von Unterricht, halten wir an einer **tendenziellen Komplementarität von Inhalt und Vermittlung** fest. Diese Komplementarität drückt sich in zahlreichen Bezügen zwischen didaktisch-methodischen Fragestellungen und Problemen der Qualifikationsanforderungen moderner Software-Entwicklung aus, die die Kapitel 4 bis 7 jeweils beschließen werden.

[86] Diese Trennung hat nicht nur pragmatische Gründe der Darstellung. Vielmehr muß der konkrete Bezug zwischen didaktischen Methoden und den Inhalten des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« jeweils nach den realen Gegebenheiten der Organisation der Ausbildung hergestellt werden. Darauf, daß die Durchsetzung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« nicht allein auf der Ebene der Unterrichtsgestaltung stattfinden kann, werden wir im Kapitel 1 des folgenden Teils ausführlicher eingehen.

III Didaktisch-methodische Umsetzung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz«

1 Einleitung

1.1 Ein Paradigmenwechsel in der Qualifizierung

Das Qualifizierungsziel »Ganzheitliche Arbeitsgestaltungskompetenz« unterscheidet sich grundlegend von partiellen Qualifizierungszielen, wie dem Beherrschen einer bestimmten Programmiersprache. Dieser Unterschied besteht einerseits bezüglich der Komplexität, andererseits bezüglich des Verhältnisses zur bisherigen Qualifizierungspraxis, zu den Qualifizierungszielen und der Unterrichtsgestaltung. Ganzheitliche Arbeitsgestaltungskompetenz kann nicht der bisherigen Qualifizierungspraxis angefügt werden, sie kann auch nicht vorhandene Qualifizierungsziele ersetzen, so wie eine neue Programmiersprache beispielsweise an die Stelle einer veralteten Programmiersprache tritt. Die Umsetzung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« erfordert eine Umgestaltung der gesamten, vorhandenen Qualifizierungspraxis. Wir sprechen daher auch von einem Paradigmenwechsel in der Qualifizierung.

Der Begriff des Paradigmenwechsels hat seinen Ursprung in der Beschreibung von wissenschaftlichem Wandel und besagt im wesentlichen folgendes: Während sich die Wissenschaft »normalerweise« durch die Erweiterung von im Kern unveränderten Theorien weiterentwickelt, bezeichnet der Paradigmenwechsel das Ersetzen alter Theoriekerne durch neue. Auf die Qualifizierung bezogen kann nun analog von der Ersetzung alter Kernqualifikationen durch neue gesprochen werden. Allerdings wollen wir einen gravierenden Unterschied, den diese Analogie verdeckt, nicht unbeachtet lassen. Der wissenschaftliche Paradigmenwechsel korrespondiert mit dem Legitimationszwang der wissenschaftlichen Arbeit, ständig neues Wissen zu produzieren und ist somit der wissenschaftlichen Arbeit inhärent.

Ein Paradigmenwechsel in der praktischen Qualifizierungsarbeit wird dagegen meist als äußerer Zwang empfunden, der die aktuelle Arbeit behindert und nicht produktiv inspiriert. Auf den Punkt gebracht heißt das: Der Wissenschaftler kann seine bisherige Arbeit liegen lassen und sich einem neuen Paradigma zuwenden. Der Ausbilder kann dies nicht. Die Arbeit des Unterrichtens geht kontinuierlich weiter, muß weitergehen. Das bedeutet für uns: Der Paradigmenwechsel, den das Qualifizierungsziel »Ganzheitliche Arbeitsgestaltungskompetenz« markiert, kann nicht in Form eines scharfen Schnitts praktisch durchgesetzt werden, sondern muß sich langsam und kontinuierlich in einzelnen Schritten entwickeln.

Im folgenden machen wir Vorschläge, wie dieser kontinuierliche Prozeß des Paradigmenwechsels in der praktischen Ausbildung aussehen könnte. Diese Vorschläge haben einen tendenziell anderen Bezugsrahmen als die bisher vorgestellten theoretischen Überlegungen: Über ihre relative Gültigkeit wird nicht in einem wissenschaftlichen Diskurs, sondern in der Praxis der Ausbildung entschieden.

1.2 Das Problem der Durchsetzung des Paradigmenwechsels

Der Umsetzung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« geht die Durchsetzung des Paradigmenwechsels bei den Dozenten voraus. Diese Durchsetzung kann nicht oktroyiert, sondern nur in einem gemeinsamen Kommunikationsprozeß entwickelt werden. Kerngedanke dieser Durchsetzung ist es, das Qualifizierungsziel »Ganzheitliche Arbeitsgestaltungskompetenz« zum allgemeingültigen Leitbild der Ausbildung zu machen. Leitbildcharakter kann das Qualifizierungsziel »Ganzheitliche Arbeitsgestaltungskompetenz« allerdings nur dann erreichen, wenn seine Plausibilität mit einer angenommenen Praktikabilität korrespondiert. Diese Praktikabilität kann wiederum letztendlich nur in der Umsetzung nachgewiesen werden. So drehen wir uns tendenziell in einem »circulus vitiosus«: Die Durchsetzung des Leitbildes ist Voraussetzung für seine Umsetzung, welche ihrerseits wiederum Voraussetzung für die Durchsetzung ist. Daß dieser Kreis keine Fiktion ist, haben wir in zahlreichen Gesprächen mit Ausbildern erfahren. Während wir auf die Plausibilität des Qualifizierungsziels insistierten, orientierten sich die Dozenten an der Praktikabilität des Qualifizierungsziels als Ganzem. Damit ist aber die Kommunikation zwischen uns (als Sozialwissenschaftlern) und den Dozenten immer von einem strukturellen Konflikt überlagert worden.

Wenn dieses strukturelle Dilemma des Nichtverstehens aufgrund unterschiedlicher Arbeitsbezüge total wäre, könnte hier auf jegliche Diskussion verzichtet werden. Dies ist freilich nicht der Fall. Trotzdem kann u.E. dieses Dilemma nicht dahingehend einfach aufgehoben werden, daß an eine jenseits der konkreten Arbeitsbezüge vorhandene allgemeine Vernunftbasis appelliert wird. Insbesondere würde dies auch unseren eigenen theoretischen Überlegungen zu den Voraussetzungen kommunikativen Handelns widersprechen. Hinter dieser Aussage steht die Frage, wie der Prozeß der Durchsetzung eines Paradigmenwechsels der Ausbildung in einem Kommunikationsprozeß zwischen Ausbildern und Wissenschaftlern organisierbar ist, und welche personalen Verhaltensweisen von den einzelnen Gruppen zu erwarten sind. Die Beantwortung dieser Frage besteht u.E. in der Herstellung eines **transparenten Kommunikationsrahmens**, der mit **spezifischen Verhaltenserwartungen** an die Kommunikationsteilnehmer verknüpft ist. Einen solchen Kommunikationsrahmen konnten wir gemeinsam mit unseren Kooperationspartnern herstellen. Prinzipien eines solchen

Kommunikationsrahmens, der die Voraussetzung für die Durchsetzung und damit auch Umsetzung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz«, wollen wir im nachfolgenden darstellen.

1.3 Kommunikationskultur und Organisationskultur als Voraussetzungen zur Umsetzung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz«

Die im folgenden dargestellten Kriterien eines Kommunikationsrahmens beziehen sich keineswegs in erster Linie auf die Kommunikation zwischen Wissenschaftlern und praktisch tätigen Ausbildern, sondern sind auch als Voraussetzung für einen selbstinitiierten Paradigmenwechsel innerhalb von Ausbildungsinstitutionen anzusehen. Wir behaupten sogar, daß Ausbildungsinstitutionen im IT-Bereich nur durch die Institutionalisierung eines solchen Kommunikationsrahmens auf Dauer den Komplexitätszuwächsen ihres Faches gerecht werden. Ansonsten laufen sie Gefahr, sowohl von der wissenschaftlichen Informatik und deren Ausbildungsleistungen als auch von den Erfordernissen der beruflichen Praxis abgekoppelt zu werden.

Der im folgenden vorgeschlagene Kommunikationsrahmen verlangt eine bestimmte demokratische Organisationskultur der Ausbildungsinstitution. Die Durchsetzung und Umsetzung des Qualifizierungsziels Ganzheitliche Arbeitsgestaltungskompetenz ist u.E. maßgeblich von der jeweiligen Organisationskultur des Ausbildungsträgers abhängig. Die Organisationskultur bezeichnet etwas, das in erster Linie hinter dem Rücken der einzelnen Akteure wirkt: Wie hierarchische Strukturen in der personalen Kommunikation bewältigt werden, wie Konflikte zwischen einzelnen Personen oder Abteilungen ausgetragen werden, wie Arbeitsprobleme nach außen dargestellt werden und vieles mehr, hängt weniger vom einzelnen ab, als vielmehr von einer Gruppenstruktur, die scheinbar unveränderlich schon immer vorhanden ist.

Der im folgenden vorgeschlagene Kommunikationsrahmen kann daher an einer Organisationskultur scheitern, die die in diesem Rahmen implizierten Kommunikationsformen konterkariert. Das heißt im Klartext: Wenn die von uns als Voraussetzung für die Ganzheitliche Arbeitsgestaltungskompetenz beschriebenen sozialen Kompetenzen in den Interaktionsbeziehungen der Organisationsmitglieder nur verzerrt zum Tragen kommen, kann kaum erwartet werden, daß sie innerhalb dieser Organisation vermittelt werden können. Wir gehen allerdings im folgenden davon aus, daß weder eine ideale noch eine total destruktive Organisationskultur vorliegt. Dies ist dann der Fall, wenn Probleme der Organisationskultur den von uns anvisierten Kommunikationsrahmen nicht sprengen, sondern in diesem konstruktiv bearbeitet werden können. So gesehen ist der durch nachfolgende Kriterien gekennzeichnete **Kommunikationsrahmen auch ein Mittel zur Verbesserung der Organisationskultur**:

Formale Kriterien:

- Der Kommunikationsrahmen steht allen Ausbildern gleichermaßen offen und ist **hierarchiefrei** organisiert.
- Die Teilnahme an der Kommunikation ist freiwillig, wird aber als **reguläre Arbeitszeit** angerechnet.
- Die Kommunikation findet **regelmäßig** statt.
- Die Ergebnisse werden dokumentiert. **Transparenz** ist eine Grundregel des gesamten Kommunikationszusammenhangs, sowohl der Plena als auch der Arbeitsgruppen.

Thematische Kriterien:

- Um den Zweck des Kommunikationsrahmens, **die Neuausrichtung der Ausbildung aufgrund von organisationsübergreifenden Informationen** (der Praxis, der Hochschulinformatik) nicht zu gefährden, muß die Kommunikation **thematisch eingeschränkt** werden:
  1. Thema der Kommunikation sind die **Erfordernisse beruflicher IT-Arbeit** und die **Entwicklungstendenzen im Bereich der Software-Entwicklung**, wie sie durch die universitäre Informatik beschrieben werden. Dazu werden regelmäßig Referenten eingeladen.
  2. Ein weiteres Thema sind **didaktische Fragestellungen**, die sich mit der Umsetzung neuer Qualifizierungsziele befassen. Hierzu werden bei Bedarf ebenfalls auswärtige Referenten angefragt.
  3. Die Kommunikationsteilnehmer verpflichten sich, alltägliche Unterrichtsprobleme nicht zum Thema der Kommunikation zu machen, wenn diese sich nicht auf prinzipielle Probleme bei der Neuausrichtung der Ausbildung beziehen.
- Um das Prinzip der Freiwilligkeit, auf dem allein ein erfolgreicher Paradigmenwechsel basieren kann, nicht zu gefährden, muß die thematische Einschränkung gleichzeitig mit Offenheit verbunden sein. Bedingung ist dafür: Die einzelnen Themen werden von allen Teilnehmern gleichberechtigt vorgeschlagen und ausgewählt. Die Auswahl der Themen folgt in der Regel dem **Prinzip des Konsens** und nicht der Mehrheit.

Zielkriterien:

- Erstes Ziel ist es, einen **verbindlichen Konsens über die Ausrichtung der zukünftigen Ausbildung** zu bilden und einzelne Problembereiche dieser Neuausrichtung zu identifizieren.

- Zweites Ziel ist es, eine oder mehrere Arbeitsgruppen einzusetzen, die diesen Konsens als **verbindliches Curriculum** ausarbeiten. Die Ergebnisse der Arbeitsgruppe werden regelmäßig innerhalb des größeren Kommunikationsrahmens präsentiert und diskutiert. Dazu werden in regelmäßigen Abständen auch Vertreter der Praxis und der universitären Informatikerausbildung eingeladen.
- Drittes Ziel ist es, **Pilotprojekte zu initiieren**, die den Anforderungen des neuen Curriculums auch auf der Ebene der Unterrichtsgestaltung gerecht werden. Der Umfang dieser Pilotprojekte kann sehr unterschiedlich sein. Bei umfangreichen Pilotprojekten ist eine didaktisch-methodische wissenschaftliche Begleitung ratsam. Auch der Verlauf der Pilotprojekte wird der allgemeinen Kommunikation zugänglich gemacht.

1.4 Unser Beitrag zur Selbstorganisation des Paradigmenwechsels

Die nachfolgenden Kapitel beinhalten Diskussionslinien und Diskussionsaspekte, die von unserer Seite zu den Themenfeldern eines solchen Kommunikationsrahmens vorgeschlagen werden. Sie beziehen sich insbesondere auf den didaktisch-methodischen Bereich der Umsetzung des Qualifizierungsziels Ganzheitliche Arbeitsgestaltungskompetenz und schließen dabei direkt an die Darstellung der Dimensionen der beruflichen Handlungskompetenz an.

Unsere Aufgabe kann es freilich nicht sein, fertige Rezepte zur didaktisch-methodischen Umsetzung dieses Qualifizierungsziels zu liefern. Dies hat zwei leicht einsehbare Gründe: Erstens, wir verfügen nicht über solche Rezepte, da diese wissenschaftlich nicht herstellbar sind. Zweitens, selbst wenn es solche Rezepte - beispielsweise in der idealtypischen Beschreibung eines Pilotprojektes - geben würde, könnten diese von einem Dozenten nicht einfach angewendet werden. Unterricht ist immer anders.

Der Anspruch, den wir hier mit unserer Arbeit verfolgen, ist allerdings keineswegs geringer, geht es uns doch darum, auf indirektem Wege **zur Selbstorganisation eines Paradigmenwechsels in der Ausbildung beizutragen**. Dieses haben wir dadurch versucht, daß wir wissenschaftlich fundierte Fragestellungen und Aspekte der didaktisch-methodischen Umsetzung ausgearbeitet haben, die wir den Ausbildern zur **Reformulierung ihrer Praxis** zur Verfügung stellen. Nur in diesem wechselseitigen kommunikativen Prozeß besteht die Möglichkeit, den wissenschaftlichen Anspruch mit dem Anspruch der Praxisveränderung zu verbinden.

Die nachfolgenden Kapitel gehen gleichermaßen auf theoriegeleitete Reflexionen und zahlreiche Gespräche mit Dozenten unserer Kooperationspartner zurück.

Das folgende Kapitel behandelt die Frage nach der **Besonderheit des pädagogischen Handelns**. Es reflektiert die Schwierigkeiten, Dozent zu sein, die Schwierigkeit, daß es keine didaktische Maschine gibt, daß der heimliche Lehrplan auch für den Dozenten heimlich bleibt und schließlich das Dilemma, daß derjenige, der unterrichtet, nicht gleichzeitig beobachten kann, was er unterrichtet. Kurz: Es geht um die besonderen **Unsicherheiten**, die das berufliche Handeln von Ausbildern virtuell immer begleiten, die aber besonders in Situationen einer paradigmatischen Umorientierung zum Problem werden, das reflektiert werden muß.

Die Kapitel 4 bis 7 nehmen die Dimensionen der beruflichen Handlungskompetenz, **Wissen, soziale Kompetenzen** und **berufliche Identität, als heuristische Modelle zur Strukturierung didaktischen Handelns** wieder auf. Zentral ist hierbei die These, daß eine ganzheitliche Umsetzung des Qualifizierungsziels Ganzheitliche Arbeitsgestaltungskompetenz die didaktische Berücksichtigung dieser drei Dimensionen zwingend benötigt. Insbesondere wird in diesem Block die praktische Relevanz dieser heuristischen Modelle anhand von Beispielen aufgezeigt. Die im ersten Block hergeleiteten Unsicherheiten pädagogischen Handelns bekommen im zweiten Block eine konstruktive Basis. Das heißt: Wir stellen **Handlungsorientierungen** vor, die diese Unsicherheiten zwar nicht leugnen, aber Handeln in tendenziell unsicheren Situationen ermöglichen.

2 Von den Schwierigkeiten, Dozent zu sein

Gegenstand der Analyse der beruflichen Handlungskompetenz ist zuerst das berufliche Tun. Erst nachdem dieses analysiert worden ist, kann es als allgemeine Zurechnung auf die Person, das heißt, als Kompetenz bestimmt werden. Wenn wir nun die **berufliche Handlungskompetenz des Lehrers** betrachten, geraten wir sehr schnell in Schwierigkeiten: "Wir gehen dabei von einem einfachen, aber grundlegenden Faktum aus, nämlich von der Tatsache, daß der Lehrer nicht in erster Linie auf seine Schüler wirkt durch das, was er *tut*, sondern durch das, was er *ist*" [Otte 1994, S. 161].

Dieser als unumstößliche Tatsache vorgebrachte Satz Ottes verweist nicht nur auf die Schwierigkeiten des handelnden Lehrers, der nur seine Handlungen planen und regulieren kann, aber sein **wirkendes Sein** nicht zum Handlungsgegenstand machen kann. Vielmehr erwachsen aus diesem Satz auch Probleme, denen sich eine wissenschaftliche Fachdidaktik dann stellen muß, wenn sie als Berufswissenschaft des Ausbilders den Anspruch praktischer Relevanz erhebt. Die Fachdidaktik kann sich nicht mehr auf eine transformatorische Position zurückziehen, die die optimale Übertragung von wissenschaftlichem Wissen in schulisches Wissen beschreibt, ohne die **Grundproblematik pädagogischen**

Handelns als personales Lehrer-Schüler-Verhältnis miteinzubeziehen. Das heißt: Die Fachdidaktik muß nicht nur ihr **Verhältnis zur Wissenschaft klären,** sondern gleichzeitig auch einen **Bezug zur Komplexität des pädagogischen Lehrer-Schüler-Verhältnisses herstellen**. "Eine Vernachlässigung dieser Besonderheit des Lehrerberufs läuft auf die Vorstellung der Didaktik als einer Methodenlehre, gewissermaßen als einer »Technologie des Nürnberger Trichters«, hinaus" [Otte 1994, S. 161]. Hierzu werden wir folgende Thesen kurz ausführen:

- Der Abschied von der **"didaktischen Maschine"** [Adam 1988, S. 157] bildet den Ausgangspunkt einer notwendigen didaktischen Umorientierung.
- Der Informatiklehrer vertritt als **"exemplarischer Intellektueller"** [Otte 1994, S. 161] die Gesamtheit des Faches und keine Programmiersprache, selbst wenn er eine Programmiersprache unterrichtet. Die personale Fachlichkeit des Lehrers ist im pädagogischen Verhältnis immer eine allgemeine.
- Der **"heimliche Lehrplan"**[87] ist auch für den Lehrer heimlich, das heißt, es besteht für ihn die Notwendigkeit der Reflexion des Handelns. Aber: Wer unterrichtet, kann sich nicht beobachten, wie er unterrichtet. Die Notwendigkeit von reflexiver Planung und Evaluation des Unterrichts bleibt demnach immer zwingend. Pädagogisches Handeln findet im Zwischenreich **"zwischen Absicht und Person"**[88] statt. Die Bemühungen einer **"Beschleunigungsdidaktik"** [Rumpf 1986, S. 105], die auf einen möglichst reibungslosen und schnellen Wissenstransfer abzielt, führen hier nicht weiter, sondern sind eher hinderlich.
- Wissen, soziale Kompetenzen und berufliche Identität sind **heuristische Modelle des Handelns im "Zwischenreich"** [Tenorth 1992, S.194]: Diese heuristischen Modelle setzen den »exemplarischen Intellektuellen« voraus. Sie ermöglichen permanent reflexive Bezüge sowohl zur fachlichen Absicht als auch zum personalen Lehrer-Schüler-Verhältnis, ohne dieses Problem der Übertragung technologisch verkürzt zu lösen.

Die im nachfolgenden explizierten Thesen dienen auch dazu, dem Ausbilder die Komplexität von Unterricht wieder bewußt zu machen. Es ist nicht zuletzt die

---

[87] Das Konzept des heimlichen Lehrplans hat Bernfeld in den Zwanziger Jahren als ideologiekritischen Begriff entwickelt. Dieses Konzept "betont die Bedeutung der institutionell geschaffenen Beziehungsmuster, vernachlässigt dabei aber völlig die Unterrichtsinhalte" [Tillmann 1989, S. 169]. Wir benutzen den Begriff des heimlichen Lehrplans hingegen als erweiterten Begriff, der auch die mit der Vermittlung von Lerninhalten einhergehenden, nicht beabsichtigten Vermittlungen von Einstellungen, Wahrnehmungsmustern usw. miteinbezieht.

[88] Diese grundsätzliche pädagogische Problematik bildet den Titel eines von Luhmann/ Schorr herausgegebenen Sammelbandes [Luhmann/Schorr 1992].

»Ungeheuerlichkeit« des Unterrichtens, absichtsvoll ein spezifisches Verstehen und bestimmte Einstellungen bei anderen in einem kommunikativen Prozeß zu schaffen, die wir uns hier wieder vor Augen führen müssen.[89]

2.1 Abschied von der didaktischen Maschine

"Analysis bedeutet ein Zerlegen des Ganzen in seine Teile, sie ist die erste und wahrhaft fundamentale Grundlage einer wirklichen Kenntnis. Was ungeteilt und ungeschieden, ist ein Durcheinander; es verwirrt die Sinne, den Geist und sich selbst. In der Unterscheidung ist Licht. (...) Eine Analysis muß daher exakt durchgeführt werden, sonst wird die betrachtende Vertiefung in die Dinge verhindert und die Erkenntnis wird trügerisch. (...) Synthesis ist die Wiederzusammensetzung der Teile zu dem ihnen entsprechenden Ganzen. Wenn sie wahrhaft ist, nützt sie gar sehr der sicheren Kenntnis der Sachenwelt. Denn aus der Beobachtung der Teile an sich entspringet kein Nutzen, und es ist nicht leicht einzusehen, wozu sie dient. Werden die Teile aber wieder in die richtige Ordnung gebracht und miteinander verbunden, zeigt sich sofort, wozu sie nutze sind, wie an der zerlegten und wieder zusammengesetzten Uhr zu sehen ist" [Comenius 1681, zit.n. Adam, S. 163].

Dieses von Comenius im 17.Jahrhundert entworfene didaktische Modell kann als das Paradigma der Didaktik schlechthin gelten. Die Zerlegung ineinander verzahnten Wissens, das eine mechanistische Wirklichkeit widerspiegelt, korrespondiert mit einem Verstand, der selbst wiederum Maschinencharakter hat. **Wirklichkeit - Wissen - Wissensvermittlung - Verstehen bilden eine Kette, die niemals abreißt, weil sie in jeder Formgestalt den mechanischen Gesetzen der Maschine folgt.** Die Analyse wird somit nie problematisch, weil sich ihre Synthese zwangsläufig und eindeutig einstellt. Wenn ein Lehrer nun einzelne, aus der Zerlegung hervorgegangene Wissensteile unterrichtet, muß er sich im Moment des Unterrichtens nicht um die Synthese sorgen. Selbst im einzelnen verkörpert **er** das Ganze. Eine moderne, gängige Form dieses Paradigmas findet sich in der lerntheoretisch ausgerichteten Didaktik wieder.

---

[89] Diese »Ungeheuerlichkeit«, etwas zu tun, das man in seinem Wesen nicht begreift, treibt die Pädagogik seit ihrem Bestehen um. Selbst die empirische didaktische Methodenforschung stößt immer wieder an ihre Grenzen, wenn sie zwar benennen kann, daß eine bestimmte Methode einer anderen in dem Sinn überlegen ist, daß sie einen reproduzierbaren Lernerfolg in kürzerer Zeit wahrscheinlicher macht, aber letztendlich über das Lernen wenig aussagen kann. (Man denke an die mit großer Vehemenz geführte Diskussion um die »Ganzwortmethode« beim Schreibenlernen und ihre unterschiedlichen Bewertungen, die durchaus als Glaubensbekenntnisse aufgefaßt werden können. Ähnliches gilt auch für die Mengenlehre. Kurz gesagt: Wir wissen sehr wenig.).

Die Vorgehensweise der lerntheoretischen Didaktik ist nicht zuletzt wegen ihrer instrumentellen Verwendbarkeit äußerst beliebt. Die Planung des Unterrichts wird analytisch in Entscheidungs- und Bedingungsfelder zerlegt, die aufeinander bezogen werden. Die Entscheidung folgt einer Fragenreihe:

- Was soll gelernt werden? (**Intentionalität**)
- Welchen Lerngegenstand benutze ich dazu? (**Thematik**)
- Welche Methode benutze ich? (**Methodik**)
- Welche Medien setze ich ein? (**Medienwahl**).

Die Entscheidung berücksichtigt folgende Bedingungen:

- Wissensstand und Lernfähigkeiten (einschl. Motivation) der Lernenden (**personale Bedingung**)
- Organisatorische Möglichkeiten (**sozialkulturelle Bedingungen**).

Darüber hinaus soll die Entscheidung noch folgenden Prinzipien gehorchen:

- Alle Planungsmomente sollen eine widerspruchsfreie Wechselwirkung haben, wobei der Intentionalität das Primat zukommt (**Prinzip der Interdependenz**).
- Das methodische Vorgehen muß angesichts unvorhersehbaren Schülerverhaltens alternative Möglichkeiten einschließen (**Prinzip der Variabilität**).
- Das Erreichen des ursprünglich intendierten Lernziels muß überprüfbar sein (**Prinzip der Kontrollierbarkeit**).

Mit diesen Planungs- und Entscheidungsmomenten liefert die lerntheoretisch orientierte Didaktik ein komplexes Modell zur Zerlegung von Wissen zum Zweck seiner unterrichtstechnologischen Verfügbarmachung. Wer dieses Modell aus prinzipiellen Überlegungen kritisiert, der muß auch angeben, welche didaktischen Richtlinien er anstelle der lerntheoretischen Prinzipien und Fragestellungen setzen will. Dieser Generaleinwand der Vertreter einer lerntheoretisch ausgerichteten Didaktik gegen ihre subjektorientierten Kritiker rekurriert allerdings auf eine unterstellte Praktikabilität der eigenen Prinzipien und Richtlinien, die durchaus anzuzweifeln ist. Das Ausblenden sowohl der Schüler als auch der Lehrer als selbsttätige Subjekte im Erziehungsprozeß führt zu ursächlichen Zuschreibungen didaktischer Erfolge, die so eindeutig nicht sind. Das heißt:

1. Die Lernleistung stellt keineswegs allein das Resultat einer an Lerninhalten orientierten Didaktik dar, sondern ist immer auch das Resultat eines pädagogischen Lehrer-Schüler-Verhältnisses.

2. Die lerntheoretisch hergestellte Lernleistung bezieht sich auf eine bestimmte Form von adaptivem Wissen und hat nur hier ihre nicht zu verallgemeinernde Berechtigung.

Wenn wir nun unsere Bildungsziele Technikgestaltungskompetenz, Prozeßkompetenz und Kooperationskompetenz näher betrachten, werden die Defizite des lerntheoretischen Paradigmas deutlich. Die Zerlegung der Ganzheitlichen Arbeitsgestaltungskompetenz in einzelne Kompetenzen folgt nicht der Logik eines an der Maschinenmetapher oder, moderner ausgedrückt, eines an Lernalgorithmen ausgerichteten Prinzips. Schon der Terminus der »Ganzheitlichkeit« weist darauf hin, daß die von uns anvisierte Arbeitsgestaltungskompetenz zwar bezüglich des situativen Charakters des Arbeitsprozesses durch einzelne Kompetenzen beschreibbar ist, aber diese Teilkompetenzen nur in einem gemeinsamen Wirkungszusammenhang gesehen werden können. Eine Didaktik der Ganzheitlichen Arbeitsgestaltungskompetenz muß ihrem Charakter als vernetztes Denken und Handeln gerecht werden. Sie muß damit Abschied nehmen von dem Paradigma der Linearität des verzahnten Wissens, das seine fest definierten Schnittstellen kennt.

Dieser notwendige Abschied vom didaktischen Maschinendenken, das auf einer erkenntnistheoretisch nicht hinterfragten, mechanistischen Strukturaffinität von Wirklichkeit, Lerninhalten, Unterrichtsmethode und Lernen beruht, bedeutet allerdings nicht, daß die oben dargestellten Entscheidungs- und Bedingungsfelder der Unterrichtsplanung wertlos geworden wären. Vielmehr erhalten sie den Charakter eines heuristischen Modells, das neben anderen heuristischen Modellen, wie der Performanz des Wissens, der Einübung sozialer Kompetenzen und dem Identitätslernen, steht. Bevor wir uns diesen anderen heuristischen Modellen zuwenden, wollen wir die Schwierigkeiten und Chancen reflektieren, die mit einer unsicher gewordenen Position des Lehrenden einhergehen. Diese Unsicherheit resultiert daraus, daß didaktisches Handeln nicht mehr auf stark strukturierte didaktische Methoden bezogen werden kann, daß es also keinen archimedischen Punkt gibt, von dem aus didaktisches Handeln eindeutig strukturiert werden kann, sondern nur noch ein diffuses personales Lehrer-Schüler-Verhältnis, das in seiner Dynamik kaum einen orientierungsstiftenden Fixpunkt darstellt.

Wir wollen uns also die Frage stellen, welche Handlungsorientierung für einen Lehrenden angemessen ist, der sein Handeln nicht mehr an der Anwendung didaktisch-technischer Methoden orientieren kann, sondern auf ein instrumentell undurchdringliches personales Lehrer-Schüler-Verhältnis verwiesen wird.

2.2 Der Lehrer als »exemplarischer Intellektueller«

"Der Lehrer erfüllt (...) seine gesellschaftliche Aufgabe nicht nur dadurch, daß er ein bestimmtes Wissen und bestimmte Techniken bewußt anwendet, sondern auch durch **seine Wirkung als Persönlichkeit**. Das Konzept des »exemplarischen Intellektuellen« ist schwer zu fassen, da es einerseits nicht im Sinn einer Reifizierung, d.h. im Sinne der Identifizierung der Psychologie einzelner Individuen, zu verstehen ist. Man kann nicht davon ausgehen, daß sich die Mehrheit der Lehrer bewußt als (exemplarische) Intellektuelle sehen. Andererseits ist dieser Begriff aber auch nicht bloß normativ aufzufassen, als gesellschaftliche Normvorstellung oder Bildungsanforderung. Es ist für den Lehrer in seinem individuellen **Selbstverständnis** wichtig, sich dieses Aspekts als eines **notwendigen integrierenden Moments seiner eigenen Gesamttätigkeit** bewußt zu werden" [Otte, S. 167, Herv. d. V.]. Wir werden dieses notwendige Selbstverständnis als adäquate berufliche Identität bezeichnen. Adäquat deshalb, weil es u.E. das einzige Selbstverständnis ist, das einen fachlichen Bezug mit einem personalen Bezug zum pädagogischen Verhältnis verbinden kann.

In den nachfolgenden Punkten, die die Position des Lehrenden als »exemplarischen Intellektuellen« sowohl aus fachlicher Sicht als auch aus pädagogischer Sicht begründen, folgen wir den Überlegungen Ottes:

- Der Lehrer hat zum Fachwissen prinzipiell nur einen **intellektuellen Zugang**, da er die Adäquatheit des fachlichen Wissens nicht aus der selbsttätigen Anwendung in praktischen Arbeitszusammenhängen schöpfen kann. Bei der Betrachtung des Fachwissens muß er sich sowohl auf das Wissenschaftssystem des spezifischen Fachwissens als auch auf die theoretisch reflektierte Berufspraxis beziehen. Des weiteren muß er von der Bedeutsamkeit des vermittelten Wissens auch selbst überzeugt sein. Das heißt: Er muß einzelne Wissensbereiche auf ein **reflektiertes, sinnorientiertes Gesamtwissen** beziehen. Verzichtet er darauf, dann bietet er Wissen nur noch als Ware feil. Wissen mit Warencharakter steht aber in diametralem Widerspruch zum Qualifizierungsziel »Ganzheitliche Arbeitsgestaltungskompetenz«, das auf die Übernahme personaler Verantwortlichkeit abzielt, welche wiederum nur durch die Reflexion komplexer, aufeinander bezogener Bezugssysteme zu leisten ist.
- Erst diese **Selbstüberzeugung** und sein Bezug auf ein sinnorientiertes Gesamtwissen ermöglicht es dem Lehrer, sein Wissen als **Anliegen** vorzutragen. Erst dieses Anliegen führt dazu, daß der Lehrer einen bestimmten **Geist** und eine bestimmte **Glaubwürdigkeit** ausstrahlt. Ein Anliegen kann freilich nicht verordnet werden. Aber wir glauben kaum, daß das Qualifizierungsziel Ganzheitliche Arbeitsgestaltung rein methodisch realisiert werden

kann, wenn es nicht zum personalen Anliegen der Lehrenden geworden ist. Das »Anliegen« bezeichnet gewissermaßen die personale Ebene des institutionellen Leitbildes, die Selbstüberzeugtheit und die personalen Konsequenzen einer Kommunikation, wie wir sie weiter oben unter dem Stichwort »Kommunikationsrahmen« skizziert haben.

- Der Geist und die Glaubwürdigkeit des Lehrers sind grundlegende personale Faktoren, die das pädagogische Lehrer-Schüler-Verhältnis nachhaltig bestimmen, das heißt, den Schüler **zum Selbstlernen motivieren** und seine Orientierungen maßgeblich und langfristig beeinflussen. Der »Geist« bezeichnet einen Umgang mit Wissen, der über das bloße Verfügen hinausgeht, einen Umgang, der das Wissen in einen größeren **identitätsstiftenden Zusammenhang** stellt. Erst diese identitätsstiftende Leistung verleiht dem Lehrenden jene positive Autorität, auf die sich die Lernenden im pädagogischen Verhältnis beziehen können. Diese identitätsstiftende kognitive Leistung auf Seiten des Lehrenden setzt aber eine spezifische Distanz - das heißt: Intellektualität - insbesondere zu isolierten Wissensbereichen voraus. Zum Beispiel: Programmiersprachen beruhen auf Formalisierungen, die als solche in ihrem Nutzen und ihren Grenzen bezüglich des Verstehens reflektiert werden sollten.[90]

Mit der Positionseinnahme des »exemplarischen Intellektuellen« sind natürlich nicht alle Probleme des didaktischen Handelns gelöst. Auch wäre es fatal, wenn mit diesem Konzept eine generelle Feindlichkeit gegenüber didaktischen Methoden verbunden werden würde. Der in der Pädagogik zeitweise emphatisch beschriebene »geborene Pädagoge« ist ebenso passé, wie die Erwartungen an eine didaktische Methode als Unterrichtsrezept vor komplexen Lernzielen gescheitert sind. Während das Leitbild des »geborenen Pädagogen« als Kapitulation vor der Komplexität pädagogischen Handelns angesehen werden kann, basieren technologisch verfügbare didaktische Methoden auf der Leugnung dieser Komplexität. Das Konzept des »exemplarischen Intellektuellen« versucht hier eine vermittelnde Position einzunehmen, indem es einerseits die Komplexität pädagogischen Handelns als Anforderung an die Persönlichkeit wahrnimmt, andererseits aber an didaktischen Methoden als heuristische Modelle zur Bewältigung dieser Komplexität festhält. Bevor wir uns weiter didaktisch-methodischen Fragestellungen widmen, wollen wir uns kurz der Problematik der Position des Dozenten in der Weiterbildung zuwenden.

---

[90] vgl. hierzu ausführlich und gut lesbar: [Siefkes 1993].

Exkurs: Der »exemplarische Intellektuelle« in der Weiterbildung

Die Position des Lehrenden als »exemplarischer Intellektueller« haben wir einmal aus seiner Position zum Fachwissen rekonstruiert, die sich dadurch auszeichnet, daß er dieses nicht mittels direkter Anwendung integrieren kann, einmal aus seiner Stellung im pädagogischen Verhältnis, in dem er praktisch als Stellvertreter eines Faches in Erscheinung tritt. Im Rahmen der Weiterbildung können wir kaum von einem pädagogischen Verhältnis sprechen. Die Integration von spezifischen Fachinhalten zu einer größeren Gesamtheit geschieht hier vielmehr durch die **Praxiserfahrung** derjenigen, die an der Weiterbildung teilnehmen. Selbst die positive Autorität des Dozenten ist fragwürdig, da die Teilnehmer oftmals über ein handlungsorientiertes Fachwissen verfügen, das in vielerlei Hinsicht dem des Dozenten überlegen sein dürfte. Dies alles stellt allerdings kein Problem für die **traditionelle Weiterbildung** dar, weil diese auf einer **partiellen Überlegenheit des Dozenten** beruht, die sich auf einen ganz begrenzten Realitätsausschnitt bezieht. Der Dozent tritt beispielsweise als Experte eines bestimmten Betriebssystems, einer speziellen Programmiertechnik oder einer spezifischen Projektplanung auf. Er verfügt über eine sachliche Autorität geringer Reichweite.

Das Qualifizierungsziel »Ganzheitliche Arbeitsgestaltungskompetenz« als Ziel der Weiterbildung stellt die auf partieller Überlegenheit beruhende Position des Dozenten grundsätzlich in Frage. Auch für ihn ist u.E. die Übernahme der Position des »exemplarischen Intellektuellen« unerläßlich, wobei wir durchaus die Schwierigkeiten der Durchsetzung dieser Position, die meist nicht mit den Erwartungen der Teilnehmer korrespondiert, sehen. Das heißt: Das für die Weiterbildung nach wie vor geltende und selten eingelöste Postulat, **zielgruppen- und erfahrungsorientiert vorzugehen**, wird durch die Einnahme der Position des »exemplarischen Intellektuellen« durch den Weiterbildenden nicht aufgehoben, aber gebrochen. Den didaktischen Ausgangspunkt der Weiterbildung des Qualifizierungsziels Ganzheitliche Arbeitsgestaltungskompetenz bilden zu einem gewichtigen Teil **Auseinandersetzungen mit Situationsinterpretationen der Teilnehmer**, die auf deren **betrieblichem Interaktionszusammenhang**[91] beruhen. Der Teilnehmer, der in ein bestimmtes "Kommunikationsfeld dauerhaft eingebunden ist, nimmt Situationen nach den in dieser Lebenswelt geltenden Kriterien wahr, definiert die Situationen (z.B. auch Lernsituationen) nach den in dieser Lebenswelt geltenden Standards und Regeln, hat eine

---

[91] Daß die Situationsinterpretationen der Teilnehmer maßgeblich auf einem betrieblichen Interaktionszusammenhang beruhen, ist eine Prämisse erfahrungs- und zielgruppenorientierter Erwachsenenbildung. Sobald die Situationsinterpretationen der Teilnehmer vom Dozenten als unumstößliche, einzig mögliche Beschreibung von Realität angesehen werden, ist jedes weitere didaktische Handeln sinnlos, da es sich auf keine reale Realität beziehen könnte.

bestimmte Meinung über wichtige und unwichtige Probleme und antizipiert einen bestimmten Spielraum für Problemlösungen" [Mollenhauer 1976, S. 38]

Die einzige positionale Möglichkeit des Dozenten, sich mit den Teilnehmern über deren Situationsinterpretationen auseinanderzusetzen, besteht darin, die Position eines »exemplarischen Intellektuellen« einzunehmen. Der Dozent verfügt nicht über den **Erlebnishorizont** der Teilnehmer, sondern nur über einen

Reflexionshorizont, mit Hilfe dessen er - vom Allgemeinen zum Spezifischen gehend - Anregungen und Fragestellungen den Teilnehmern zur Verfügung stellen kann. Die Qualität des Reflexionshorizontes des Lehrenden kann sich niemals aufgrund vorgängiger, singulärer beruflicher Erfahrungen ergeben, sondern beruht **a priori** auf einer intellektuellen Leistung, das heißt, auf einer **reflektierten** Situationsinterpretation und deren Rückbezug auf einen **wissenschaftlichen Diskurs**. Auch in der Weiterbildung bezeichnet das Konzept des »exemplarischen Intellektuellen« eine vermittelnde Position zwischen dem Erfahrungshorizont der Teilnehmer einerseits und heuristischen didaktischen Modellen sowie wissenschaftlichen Reflexionen andererseits.

Im nächsten Abschnitt werden wir uns nochmals der Komplexität pädagogischen Handelns versichern, um in einem weiteren Schritt die Notwendigkeit der heuristischen Modelle herauszustellen, deren sich ein Lehrer als »exemplarischer Intellektueller« bedienen kann.

2.3 Das Paradigma der »Beschleunigungsdidaktik«[92] und ihr »heimlicher Lehrplan«

Ob die Schüler »etwas mitbekommen haben« oder etwas anderes gelernt haben, als man eigentlich beabsichtigt hat, bleibt während des Unterrichtens oftmals im Dunkeln. Dieses Problem wurde in der Didaktik traditionell dahingehend minimiert, daß die Lernschritte möglichst klein gehalten wurden, was eine fast **permanente Evaluation des Lernerfolges** sicherstellen konnte. Unterrichtsplanung, -durchführung und -evalution konnten so aufs engste verknüpft werden. Die didaktisch motivierte Zersplitterung von Wissen in Lernschritte erfährt duch die zeitliche Struktur des Unterrichts, wie er in vielen Ausbildungsinstitutionen festgeschrieben ist, auch einen institutionellen Zwang, der wiederum zu ihrer Legitimation herangezogen werden kann. Das pädagogische Problem des **Handelns zwischen Absicht und Person**[93] wird somit durch die Zerlegung des Wissens faktisch aufgelöst. **Die Bildungsabsicht wird auf den Lernschritt minimiert, die Person des Lernenden wird auf die Optimierung der Lernzeit reduziert.**

---

[92] Wir benutzen diesen Begriff hier durchaus als polemischen Begriff, obwohl er auch analytische Verwendung finden kann: Er bezeichnet eine didaktisch-technologische Orientierung, die sich als eigenes Ziel die Beschleunigung des gesellschaftlichen Wissenstransfers in die Köpfe der Lernenden gesetzt hat. Daß dieses Ziel direkt einem gesamtwirtschaftlichen Verwertungsinteresse entspringt, ist letztendlich der Grund dafür, daß es sich trotz massiver Kritik paradigmatisch durchsetzen konnte.

[93] Eine theoretisch ausgerichtete Diskussion pädagogischen Handelns als Handeln im »Zwischenreich« liefert Tenorth [vgl. Tenorth 1992].

Da der Lernerfolg als solcher schon über die Lerninhalte definiert und damit unproblematisch ist, bleibt das einzig Ungewisse die Zeit, die der Lernende zur Aneignung des Stoffes benötigt. Rumpf spricht in diesem Zusammenhang auch von einer **»Beschleunigungsdidaktik«**, deren Ziel es sei, die Fremdheit der Unterrichtsgegenstände möglichst rasch und möglichst restlos zu vernichten [Rumpf 1986, S. 105]. **Diese Beschleunigungsdidaktik sieht die Differenz zwischen Absicht und Person nur als Friktion im didaktischen Getriebe an.** Ihr Fachjargon ist dann auch mit der Sprache der elektronischen Datenverarbeitung kommensurabel. Es finden sich Hinweise zur "Steuerung der Geschwindigkeit der Informationsverarbeitung" [Kluwe zit.n. Rumpf 1986, S. 106], aber keinerlei Hinweise darauf, daß Lernen auch Erkennen heißt. Die Beschleunigungsdidaktik fühlt sich einem didaktischen Prinzip der Effizienz verpflichtet, das sie jederzeit mit der Menge des Stoffes und den institutionellen Sachzwängen begrenzter Lehr- und Lernzeiten legitimieren kann.

Ein Paradigmenwechsel in der Ausbildung von IT-Fachkräften, der das Qualifizierungsziel »Ganzheitliche Arbeitsgestaltungskompetenz« in den Mittelpunkt seiner Bemühungen stellt, gerät mit dem handlungsorientierenden Paradigma der Beschleunigungsdidaktik in Konflikt. Dieser Konflikt ist ein doppelter:

1. Das Qualifizierungsziel »Ganzheitliche Arbeitsgestaltungskompetenz« läßt sich nur in einem pädagogischen Verhältnis realisieren, in dem **Lernen bewußt als Selbstlernen und Erkennen** in einem kommunikativen Prozeß konzipiert wird. Ein reflektierter Umgang mit Wissen, das Einüben von sozialen Kompetenzen und das kommunikative Identitätslernen benötigen als didaktischen Bezug einen zwischen der Bildungsabsicht und der Person angesiedelten Handlungsraum (das pädagogische Verhältnis), ohne auf didaktisch-technologische Verkürzungen zurückgreifen zu können. Dieser Handlungsraum wird aber von der Beschleunigungsdidaktik systematisch ausgegrenzt und entwertet. Mit einer solchen Ausgrenzung und Entwertung wird beispielsweise oftmals auch das Fach »Informatik und Gesellschaft« konfrontiert. Die Provokation des Qualifizierungsziels Ganzheitliche Arbeitsgestaltungskompetenz besteht nun aber genau darin, daß es sich dieser Ausgrenzung widersetzt, indem es seinen Anspruch als Kernqualifikation einfordert. Die Umsetzung des Paradigmenwechsels auf der Ebene des Ausbildungsziels impliziert somit auch einen Paradigmenwechsel in der didaktischen Handlungsorientierung.
2. Die mit der Beschleunigungsdidaktik verbundenen didaktischen Handlungen erzeugen nichtintendierte Nebenfolgen, die dem Qualifizierungsziel »Ganzheitliche Arbeitsgestaltungskompetenz« fundamental widersprechen. Dieser **»heimliche« Lehrplan** führt zu einem **nomologischen und monologischen Umgang mit Wissen**, **eingeschränkten sozialen Kompetenzen**

und einer **nichtreflexiven, konditionierten beruflichen Identität**. Hauptmotor dieses »heimlichen« Lehrplans ist das Bemühen, den Bildungsinhalten schon in ihrer Performanz jegliche Fremdheit zu nehmen. Alle Probleme treten schon als von vornherein gelöste Probleme in Erscheinung. Der Fragehorizont wird systematisch beschnitten. Der »fremde Blick«, der seit Hegel - und trotz aller Kritik nach wie vor - als Urgrund authentischen Erkennens gilt, wird durch die Konsumtion von Fertigwissen abgelöst. Die Antwort steht faktisch schon vor der Frage im Raum, die ohnehin nur didaktisch-rhetorische Funktion hat.[94]

Zusammenfassend läßt sich sagen: Das Qualifizierungsziel »Ganzheitliche Arbeitsgestaltungskompetenz« ist nicht über die Handlungsorientierungen der Beschleunigungsdidaktik herzustellen. Und: Der über die Unterrichtsgestaltung der Beschleunigungsdidaktik transportierte »heimliche Lehrplan« wirkt dem Qualifizierungsziel »Ganzheitliche Arbeitsgestaltungskompetenz« aktiv entgegen.

2.4 Heuristische Modelle als Hilfen der Handlungsorientierung in einer durch Unsicherheiten geprägten Situation

Unsere bisherige Spurensuche nach der didaktischen Umsetzung des Qualifizierungsziels Ganzheitliche Arbeitsgestaltungskompetenz bestand hauptsächlich in der Form einer negativen Kritik.

1. Wir haben uns von dem Modell der "didaktischen Maschine", die nur linear verzahntem Wissen, aber nicht vernetztem Wissen gerecht wird, grundsätzlich verabschiedet.
2. Wir haben die berufliche Rolle und Identität des Lehrenden als »exemplarischem Intellektuellen«, der sein Wissen als Anliegen seiner Persönlichkeit im pädagogischen Verhältnis vertritt, neu bestimmt. Diese Neubestimmung bleibt allerdings als Handlungsmodell[95] diffus und ist haupt-

---

[94] Das Argument der Beschleunigungsdidaktiker, daß es in einer komplexen Welt nicht möglich ist, »das Rad jedesmal neu zu erfinden«, hat keine Berechtigung, wenn einer Praxis das Wort geredet wird, die jegliches »Erfinden« ausschließt. Gerade die Entwicklung von Softwaresystemen setzt ihre jeweilige »Neuerfindung« voraus. Zudem hieße es, die hier vorgetragene Kritik durch Übersteigerung zu desavouieren: Wir kritisieren nicht die Notwendigkeit didaktisch-technologischer Effizienzbemühungen, sondern ihr Totalitätsstreben, das authentische Erfahrungen aus dem Unterricht ausschließt [vgl. Rumpf 1986].

[95] Eine funktionale Neubestimmung der betrieblichen Ausbilderrolle ist unter dem Stichwort vom »Unterweiser zum Lernberater« bekannt geworden [vgl. Bojanowski 1991, S. 156]. Dieses programmatische Stichwort ist zwar griffig, hat aber keinerlei

sächlich eine negative Abgrenzung vom Wissensvermittler, der zum Wissen ein Warenverhältnis einnimmt.

3. Wir haben die vorhandene Didaktik im Kern als Beschleunigungsdidaktik kritisiert, die einem heimlichen Lehrplan Vorschub leistet, der den Bedürfnissen und Notwendigkeiten authentischer Wissenserfahrung (Erkennen) nicht gerecht wird. Damit haben wir gewissermaßen die Kritik des Modells der didaktischen Maschine um eine subjektorientierte Kritik ergänzt.

Im folgenden wollen wir dagegen positive Handlungsorientierungen **begründen**, die bei der Umsetzung des Qualifizierungsziels Ganzheitliche Arbeitsgestaltungskompetenz eingesetzt werden können. Da diese positiven Handlungsorientierungen keine Rezepte didaktischen Handelns darstellen, sprechen wir von heuristischen Modellen. Die entscheidende Idee zur Begründung positiver didaktischer Handlungsorientierungen haben wir ausführlich mit dem Begriff der beruflichen Handlungskompetenz und seinen Dimensionen Wissen, soziale Kompetenzen und berufliche Identität dargelegt. **Berufliche Handlungskompetenz kann somit als Schlüsselbegriff einer subjektorientierten Didaktik gelten.** Diese Schlüsselstellung erhält der Begriff durch seine transformatorische Leistung, die darin besteht, daß er die Anforderungen des Arbeitsprozesses in subjekt- und unterrichtszugängliche Kategorien übersetzbar macht. Der Begriff der beruflichen Handlungskompetenz ist auf beiden Seiten anschlußfähig, auf der Seite des Arbeitsprozesses und auf der Seite des lernenden Individuums. Gleichzeitig läßt er sich mit didaktisch-methodischen Modellen der Unterrichtsgestaltung verbinden.

Da wir die Relevanz der Dimensionen der beruflichen Handlungskompetenz für den Arbeitsprozeß und die subjektive Handlungsfähigkeit des Einzelnen schon hinreichend geklärt haben, wenden wir uns nun ihrer Relevanz bei der Gestaltung von Unterricht durch die Dozenten[96] zu. Als einfachstes Modell des Unterrichts schlagen wir folgendes, auf Loch[97] zurückgehendes Modell vor: Im Unter-

---

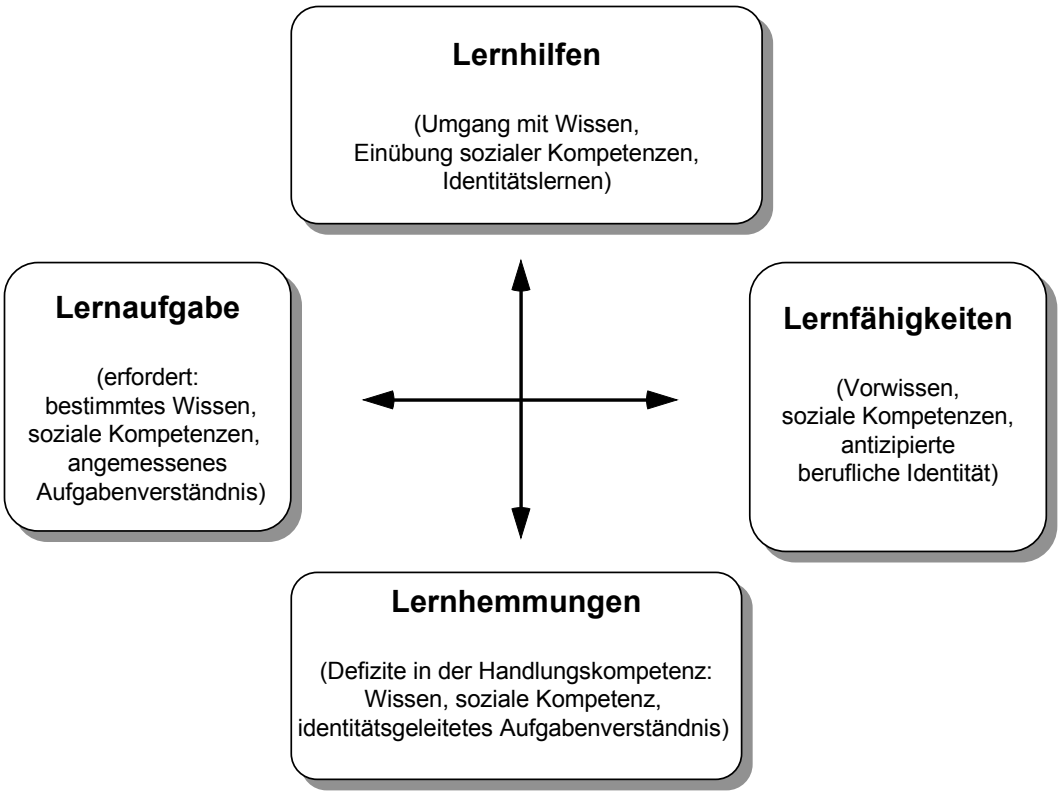
Nutzen für das Begreifen des pädagogischen Verhältnisses, in dem diese »Beratung« stattfinden soll.

96 Jeder Dozent weiß natürlich, daß der tatsächliche Unterricht nie von ihm alleine gestaltet wird, sondern eine eigene Dynamik gewinnt. Der Sinn der Gestaltung von Unterricht liegt u.E. auch nicht darin, den Unterricht methodisch voll unter Kontrolle zu bringen. Vielmehr bedeutet eine gut geplante Unterrichtsgestaltung oftmals, daß der Unterricht ganz anders als geplant verlaufen kann, ohne daß das personale Anliegen des Dozenten damit zerstört wird.

97 Loch bezieht sich allerdings nicht auf einen konkreten Unterricht, sondern beschreibt eine anthropologisch begründete biographische Erziehungstheorie [Alheit 1990, S. 25ff.].

richt werden die **Lernfähigkeiten** (Kompetenzen) der Schüler mit **Lernaufgaben** (Situationen) konfrontiert. In dieser Konfrontation treten Konflikte auf, die als **Lernhemmungen** bezeichnet werden können. Zur Bewältigung dieser Lernhemmungen stellt der Dozent **Lehrhilfen** zur Verfügung. Loch bezeichnet dies auch als »Kreuz der Erziehung«. Eine Verknüpfung dieses Kreuzes mit den von uns bestimmten Dimensionen der beruflichen Handlungskompetenzen ergibt folgendes Modell:

Didaktische Aufgaben im didaktischen Modell »Kreuz der Erziehung«



Bei näherer Betrachtung dieses didaktischen Modells können wir die didaktischen Aufgaben des Lehrers in zwei Hauptgruppen einteilen:

- Die **Gestaltungsaufgaben**, die sich auf die Planung, Durchführung und Evaluation der **Lernaufgaben** und **Lernhilfen** beziehen.
- Die **analytischen Aufgaben**, die sich auf die Analyse von **Lernhemmungen** und **Lernfähigkeiten** der Schüler beziehen.

Die Differenz von Gestaltungsaufgaben, wie dem Planen einer Lernaufgabe und dem Bereitstellen von Lernhilfen, und analytischen Aufgaben, folgt in der didaktischen Praxis oftmals der Regel, daß beide Aufgabenbereiche getrennt bear-

beitet werden. Der Lehrer gestaltet Lernaufgaben und Lernhilfen in einem gemeinsamen Gestaltungsprozeß. Lernhilfen werden aufgrund von »objektiven« Anforderungen der Lernaufgabe konstruiert, ohne daß die »subjektiven« Lernhemmungen gesehen werden. Das »erzieherische Kreuz« impliziert dagegen ein anderes Verhältnis der einzelnen Lernbegriffe: Lernaufgaben korrespondieren mit Lernfähigkeiten, Lernhilfen mit Lernhemmungen. Jeder Gestaltungsaufgabe ist eine dominierende analytische Aufgabe zugeordnet.

2.4.1 Lernaufgaben und Lernfähigkeiten

Die Lernaufgaben stehen zwischen der Analyse der Erfordernisse des Arbeitsprozesses und den Lernfähigkeiten der Schüler. Wie sich allerdings schnell zeigen läßt, gibt es eine Vielzahl von Lernaufgaben, die sich nicht direkt aus der Analyse des Arbeitsprozesses selbst ableiten lassen. Eine bestimmte mathematische Aufgabe kann durchaus als Lernaufgabe sinnvoll sein, obwohl ihre Beherrschung für das kompetente Handeln im Arbeitsprozeß vollständig irrelevant zu sein scheint. Der Grundgedanke ist hier der, daß das Ziel einer Lernaufgabe nicht allein oder gar wesentlich in der Beherrschung dieser konkreten Aufgabe liegt, sondern **daß der Prozeß der Lösung der Aufgabe Bildungsprozesse in Gang setzt, die über das konkret gestellte Aufgabenziel hinausgehen.** Das heißt: Es gibt Lernaufgaben, die nichts mit dem konkreten Arbeitsprozeß zu tun haben, aber dennoch zur Steigerung der beruflichen Handlungskompetenz beitragen. Umgekehrt gilt freilich auch, daß viele Lernaufgaben, die sich scheinbar direkt auf den Arbeitsprozeß beziehen, die Ausbildung von beruflicher Handlungskompetenz behindern. Zur Bildung adäquater Lernaufgaben kann u. E. daher nur auf das Modell der beruflichen Handlungskompetenz und ihrer Dimensionen zurückgegriffen werden. Erst die Reformulierung des übergeordneten Ausbildungsziels der Ganzheitlichen Arbeitsgestaltungskompetenz in den Dimensionen Wissen, soziale Kompetenzen und berufliche Identität kann zum Entwurf angemessener Lernaufgaben führen.

Trotz dieser Reformulierung bleibt der Arbeitsprozeß etwas dem Unterricht Äußerliches. Dies gilt selbst dann, wenn auf Erfahrungen aus dem Praktikum zurückgegriffen wird. Für das Ziel, daß die Lernaufgabe auch von den Schülern als Aufgabe betrachtet wird, die sie lösen wollen und nicht nur müssen, steht letztendlich der Dozent selbst mit seiner Person ein. Die Gestaltung einer erfolgreichen Lernaufgabe ist somit gleichermaßen das Ergebnis einer **Analyse mit Hilfe des heuristischen Modells der Dimensionen der beruflichen Handlungskompetenz**, als auch Ausdruck des authentischen Anliegens des Dozenten.

Die **Adäquatheit der Lernaufgabe bezüglich der Lernfähigkeiten** der Schüler ist eine didaktische Binsenweisheit, die zwischen den Polen Über- und Unterforderungen ihren goldenen Mittelweg sucht. Ähnlich gelagert finden wir dieses Problem der Abstimmung von Lernaufgabe und Lernfähigkeiten auch in der Weiterbildung unter den Begriffen Erfahrungs- und Zielgruppenorientierung wieder. Mit den Begriffen Wissen, soziale Kompetenzen und berufliche Identität steht ein heuristisches Modell zur Verfügung, das gleichermaßen die Lernaufgabe wie die Lernfähigkeiten beschreibbar und vergleichbar macht und somit eine differenziertere Bestimmung möglicher Über- und Unterforderung liefert.

2.4.2 Lernhilfen und Lernhemmungen

Lernhilfen beziehen sich in erster Linie auf Lernhemmungen und sind nicht sachlogisch jeweils einer Lernaufgabe zugeordnet. Ziel der Gestaltung von Lernhilfen ist es nicht, die Differenz zwischen Lernaufgaben und Lernfähigkeiten so schnell wie möglich zu überbrücken, sondern Lernhemmungen in Lernprozesse zu verwandeln.

Zentraler Ausgangspunkt unserer Überlegungen ist die **Neubestimmung von Lernhemmungen als Voraussetzung von Lernen im Unterricht.** Der Mensch stellt sich ständig Aufgaben, deren Bewältigung er sich lernend aneignen muß. Wenn er diese Aufgaben aufgrund mangelnder kognitiver Fähigkeiten, fehlender sozialer Kompetenzen oder einer unangemessenen Welt- und Selbstsicht nicht lösen kann, kommt es zu einem Konflikt zwischen Lernaufgabe und Lernfähigkeiten. Dieser Konflikt zeigt sich beim Individuum in Form von Lernhemmungen, die sich sehr unterschiedlich auf dessen Verhalten und Einstellungen niederschlagen. Die Verhaltensreaktionen gehen von resignativ bis aggressiv, die Einstellungsveränderungen schwanken zwischen Ablehnung der Aufgabe und Selbstablehnung.

Das didaktische Unterrichtsideal besteht nun darin, daß der Schüler bei auftretenden Lernhemmungen die angebotenen Lernhilfen annimmt und selbsttätig einen Lernprozeß in Gang setzt. Dies setzt voraus, daß erstens die angebotenen Lernhilfen zur Überwindung der Lernhemmungen geeignet sind, und daß zweitens die Inanspruchnahme subjektorientierter Lernhilfen sozial erwünscht ist. Beides ist keineswegs die Normalität eines Unterrichts, bei dessen Gestaltung nach wie vor von einer Kongruenz von Lernaufgabe und Lernhilfen ausgegangen und auf subjektorientierte Lernhilfen verzichtet wird.

Betrachten wir zum Beispiel die Lernaufgabe eines bestimmten mathematischen Theorems: Allgemeiner Konsens ist, daß mathematische Theoreme nicht auswendig gelernt werden sollten, da ein pures Auswendiglernen keine adäquate

Anwendung des Theorems gewährleistet. Die Lernaufgabe heißt also »Verstehen und Kennen des Theorems«, um es anwenden zu können. Das Verstehen basiert auf dem mathematischen Beweis, der sich wiederum auf schon bewiesene Theoreme, d.h. letztendlich auf Axiome stützt. Die Darstellung des mathematischen Beweises folgt einem Formalismus, der den Beweis als zwingend darstellt. Für den, der den Beweis verstanden hat, ist es unverständlich, warum andere den Beweis nicht verstehen. Er wiederholt den Beweis, betont die Stelle, an der die Bedeutung der Beweisführung evident zum Tragen kommt. Die Lernhemmungen einiger Schüler bleiben dem Lehrenden gänzlich unverständlich. Er kann sie nicht verstehen und verfügt auch über keine Sprache, mit der er über diese Lernhemmungen mit den Schülern sprechen könnte. Der Lehrende orientiert sich didaktisch allein an der Lernaufgabe, ohne daß er in der Lage wäre, adäquate Lernhilfen zu bieten. Solange der Lehrende nicht selbst auf eine "Rhetorik der Mathematik" [Siefkes 1993][98] oder auf die "Komplementarität" [Otte 1994][99] von Anschauung und Begriff zurückgreifen kann, wird er diesen unendlichen Kreis der sich wiederholenden Darstellung des Beweises niemals verlassen. Das Bemühen um eine »Rhetorik der Mathematik« und um die komplementäre Darstellung von Mathematik als Anschauung und Begriff setzt wiederum ein **reflektiertes Verhältnis zu den Gegenständen des eigenen Faches** voraus. Adäquate Lernhilfen greifen auf eine intellektuelle Auseinandersetzung des Lehrenden mit den Lernaufgaben zurück.

Im Gegensatz zur Gestaltung der Lernaufgabe, die zum größten Teil als Planung dem eigentlichen Unterricht vorausgeht, werden die Lernhilfen primär in der Interaktion mit den Lernenden im Unterricht gestaltet. Damit ist den Lernhilfen neben einem Moment der Planung auch ein Moment der Spontaneität zu eigen. Die Qualität der **spontanen Lernhilfen** beruht auf dem Erkennen der Ursachen von Lernhemmungen im aktuellen Interaktionsbezug. Neben dem Verfügen über **heuristische Modelle des Erkennens von Lernhemmungen** ist es der

---

98 Auf unterschiedliche Formen des Beweisens und ihre Problematik geht Siefkes ein [vgl. Siefkes 1993, S. 93ff.].

99 Otte weist darauf hin, daß diese Komplementarität von Begriff und Anschauungen, von Allgemeinem und Spezifischem, von Abstraktem und Konkretem für das Fach der Mathematik keineswegs nur von didaktischer Bedeutung ist, sondern die Mathematik auch wissenschaftsphilosophisch bestimmt. "Wissenschaftsphilosophie und Didaktik haben nicht nur das Anliegen gemeinsam - das Verhältnis des einzelnen zur Welt und zum Wissen über diese Welt -, sondern zeigen auch beide die Tendenz einer Behandlung ihres Gegenstandes sowohl auf eine sehr praktische, direkte Art und Weise (...) als auch in einer sehr grundsätzlichen und zuweilen abstrakt-spekulativen Form der Argumentation" [Otte 1994, S. 7].

Interaktionsbezug, sprich, das **pädagogische Verhältnis** selbst, das über die Angemessenheit von Lernhilfen entscheidet.

Die Verbindung des vorgestellten didaktischen Modells, das verschiedene Aspekte des Lernens im Unterricht erfaßt, ohne sie technologisch zu verkürzen, mit dem von uns entwickelten Modell der beruflichen Handlungskompetenz, ermöglicht eine theoretische Fundierung einer Didaktik der Ganzheitlichen Arbeitsgestaltungskompetenz. Diese theoretische Fundierung und ihre praktischen Implikationen sind Gegenstand des nächsten Kapitels.

3 Ganzheitliche Arbeitsgestaltungskompetenz und die didaktische Gestaltung des Unterrichts

Negation als Ausgangspunkt didaktischen Handelns[100]

Wer sich Ganzheitlichkeit als Qualifizierungsziel setzt, wird bald vor die Frage gestellt, in welcher Form sich diese Ganzheitlichkeit didaktisch widerspiegelt. Die Position des »holistic view« wird spätestens bei der konkreten didaktischen Umsetzung als einzig handlungsorientierendes Paradigma unhaltbar: Können die Qualifizierungsanforderungen des Arbeitsprozesses noch als »ganzheitlich« in dem Sinne beschrieben werden, daß unterschiedliche Kompetenzen additiv aufgezählt werden und auf ihre Vernetzung hingewiesen wird, so sperrt sich die didaktische Umsetzung gänzlich diesem Verfahren. **Unterrichten ist immer auf spezifische Lernprozesse fokussiert.** Unterricht bedeutet immer die absichtsvolle Fokussierung der Aufmerksamkeit der Lernenden auf einen Gegenstand. Selbst wenn dieser insgesamt multiperspektivisch betrachtet wird, so gilt doch für jede einzelne Lernsituation, daß nur jeweils eine Perspektive eingenommen wird. Auch die dezidierte Problematisierung einer ganzheitlichen Betrachtungsweise ist nicht als solche ganzheitlich, sondern macht »Ganzheitlichkeit« nur zu ihrem **einen** Gegenstand.

Wenn das Qualifizierungsziel »Ganzheitliche Arbeitsgestaltungskompetenz« auch nicht direkt in einer bestimmten Unterrichtspraxis abgebildet werden kann,

---

[100] Es ist u.a. die Notwendigkeit der Negation, die nach Otte zu strukturellen Affinitäten zwischen Fachdidaktik und Wissenschaftsphilosophie führt [vgl. Otte 1994]. Der Fachdidaktiker fragt sich ebenso wie der Wissenschaftsphilosoph, welche Widerstände des Erkennens der didaktische Gegenstand »liefert«. Eine Umgehung dieser Widerstände erzeugt nur scheinbar höhere Verständlichkeit. So zeigt Otte z.B., wie eine scheinbar verständlichere didaktische Einführung des Begriffs »kommensurable Strecken« versagen muß, weil sie den ursächlichen, negativen Begriff der »Inkommensurabilität« überhaupt nicht erfaßt, indem sie von vornherein die Streckenverhältnisse durch Verhältnisse von Dezimalzahlen ersetzt hat [vgl. Otte 1994, S. 323f].

so heißt dies keineswegs, daß es für die Gestaltung des Unterrichts nicht relevant ist. Vielmehr geht es um die Frage, wie Ganzheitliche Arbeitsgestaltungskompetenz indirekt im Unterricht umgesetzt werden kann, oder anders formuliert, wie die Erreichung des Qualifizierungsziels didaktisch-methodisch unterstützt werden kann.

Die folgenden Vorschläge zielen nun gerade darauf ab, Ansatzpunkte zu entwickeln, die die Entwicklung Ganzheitlicher Arbeitsgestaltungskompetenz dadurch ermöglichen, daß sie traditionelle Formen des Unterrichts überwinden, die diesem Qualifizierungsziel diametral entgegen stehen. Der Vorteil dieser negativen Vorgehensweise, die Friktionen benennt, die der Durchsetzung des Qualifizierungsziels Ganzheitliche Arbeitsgestaltungskompetenz entgegenwirken, hat theoretische und praktische Gründe:

- Das Qualifizierungsziel »Ganzheitliche Arbeitsgestaltungskompetenz« markiert einen **Paradigmenwechsel**. Im Gegensatz zum Paradigmenwechsel wissenschaftlicher Theorieproduktion sind der Neugestaltung von Unterricht allerdings durch starke Elemente der Kontinuität Grenzen gesetzt. Die Rollenerwartungen von Lehrenden und Lernenden, die organisatorischen Rahmenbedingungen sowie der Zwang, daß Unterricht als kontinuierlicher Prozeß immer weitergehen muß, machen es unmöglich, daß der Paradigmenwechsel als radikaler Schnitt vollzogen werden kann. Vielmehr ist es genau diese Schnittstelle zwischen Altem und Neuem, in der notwendige Veränderungen als **Kritik an bestehenden Formen der Unterrichtsgestaltung operationalisierbar** werden.
- **Die Begründung einer Didaktik ist erst über die Negation ihrer Bildungsziele praktisch fundierbar.** Ich kann zwar das Qualifizierungsziel »Ganzheitliche Arbeitsgestaltungskompetenz« als Leitbild angeben, aber erst wenn ich analysiert habe, welche Friktionen diesem auf Seite des lernenden Subjekts entgegentreten können, kann ich die subjektiven Lernhemmungen bestimmen, auf die ich mein didaktisches Handeln beziehe. Grundlage dieser didaktischen Auffassung ist die Einsicht, daß angeleitetes Lernen sich immer auf Lernhemmungen beziehen muß. Lernhemmungen sind der einzig mögliche Ausgangspunkt didaktischen Handelns. Die Unterstützung des Lernenden bei der Überwindung dieser Lernhemmungen geschieht durch Lernhilfen. Lernhilfen gehen als solche nicht aus einer übergeordneten Lernaufgabe hervor, sondern aus deren Negation.

Im folgenden gehen wir von einer **Systematik der Lernhemmungen** aus, wie wir sie mit den **Begriffen Wissen, soziale Kompetenzen und berufliche Identität** entwickelt haben. Gleichzeitig beziehen wir sie exemplarisch auf Lernaufgaben,

die der Analyse des Arbeitsprozesses folgen, also auf **Lernaufgaben aus den Bereichen der Technikgestaltung, der Prozeßgestaltung und der Kooperationsfähigkeit**. Am Schluß der jeweiligen Erörterung stehen **didaktische Prinzipien** und **Vorschläge**, die eine konkrete Umsetzung unterstützen.

4 Didaktisches Handeln und Wissen

Die Formen des Wissens sind, sowohl was ihren subjektiven Gebrauch als auch ihren gesellschaftlichen Geltungsbereich angeht, derart mannigfaltig, daß eine kohärente Systematisierung äußerst zweifelhaft ist. Dies macht schon ein assoziativer Blick auf die Formen des Wissens deutlich: Anwendungswissen, Erschließungswissen, heuristisches Wissen, Reflexionswissen, Integrationswissen, Alltagswissen, wissenschaftliches Wissen, pragmatisches Wissen, normatives Wissen usw. sind nur um den Preis willkürlicher Zwangsanwendung in ein kohärentes System zu bringen. Didaktisches Handeln kann sich demnach kaum auf eine Systematik der Wissensformen an sich beziehen. Vielmehr geht es darum, einen **systematischen didaktischen Bezug** zu den unterschiedlichen Wissensformen zu entwickeln.

Das klassische Leitbild der Allgemeinbildung ist der historisch avancierteste Versuch, ein didaktisches Verhältnis zu den unterschiedlichen Formen des Wissens herzustellen. Im Zentrum dieses Versuchs steht der Rückbezug des Wissens auf die Persönlichkeitsbildung. Die Auseinandersetzung darüber, über welches Wissen eine allseits entwickelte Persönlichkeit verfügen muß, hat seit Humboldt Legionen von Pädagogen und Bildungsplanern beschäftigt. Als ein jenseits der Person zweckfreies Wissen prägt das Humboldtsche Bildungsideal gerade auch heute die Diskussion um den Bildungsauftrag der Universitäten. Im Gegensatz zum Humboldtschen Ideal hatte und hat die berufliche Ausbildung immer ein **instrumentelles Verhältnis** zum Wissen. Dieses instrumentelle Verhältnis zum Wissen als Ausgangspunkt berufspädagogischen Handelns hat dazu geführt, daß die berufliche Ausbildung insgesamt in den Augen vieler Pädagogen als grundlegend instrumentell determiniert und wenig entwicklungsfähig angesehen worden ist. Der von uns im folgenden gemachte Vorschlag des didaktischen Umgangs mit Wissen in der beruflichen Ausbildung zeigt hingegen, daß gerade ein instrumentell orientiertes Wissensverständnis unter den sich ändernden Qualifikationsanforderungen des Arbeitsprozesses eine subjektorientierte Didaktik benötigt.[101]

---

[101] Wir gehen allerdings nicht so weit wie Bojanowski u.a., die behaupten und hoffen, daß neue Qualifikationserfordernisse generell den instrumentellen Bezug beruflicher Bildung in Frage stellen könnten [vgl. Bojanowski 1991]. "Qualifizierung als Persön-

Didaktisches Handeln in der beruflichen Ausbildung versteht Wissensvermittlung nicht als Selbstzweck, sondern als Lernhilfen, um **Widersprüche zwischen »subjektiver« und »objektiver« Wirklichkeit bei den Lernenden auszuräumen**, d.h., um unangemessene, letztendlich auf Handlungen bezogene Problemlösungsstrategien zu überwinden. Aus diesem instrumentellen Wissensverständnis ergeben sich drei didaktische Grundfragen, die die Inhalte und die Form des didaktischen Umgangs mit Wissen maßgeblich bestimmen:

1. Die Frage nach der Gestaltung der Lernaufgabe: **Auf welches berufliche Handlungsproblem bezieht sich das zu lernende Wissen?** Von welchen Lernfähigkeiten ist bei den Schülern bezüglich der Lernaufgabe auszugehen? Die Auswahl der Lernaufgabe geschieht in dem doppelten Bezug zum realen Arbeitsprozeß und den vorhandenen Lernfähigkeiten.
2. Die »objektive« Frage: **Welche Art von Wissen wird für eine adäquate Problemlösung benötigt?** Handelt es sich um ein Interpolationsproblem, ein Syntheseproblem oder ein dialektisches Problem (vgl. unsere Ausführungen zum Prozeß des Wissens)? Das heißt: Welche Rolle spielt algorithmisierbares und heuristisches Wissen bei der Problemlösung, und welche Formen der Aktualisierung und Modifikation von Wissen sind dazu notwendig?
3. Die »subjektive« Frage: **Welche Lernhemmungen sind bei den Lernenden vorhanden, die ihnen das angemessene Lösen des Problems versperren?** Fehlt es ihnen an notwendigem algorithmisierbarem Wissen oder an heuristischem Wissen? Oder liegt eine unangemessene Form der Wissensaktualisierung beziehungsweise -modifikation vor?

Aus diesen Fragestellungen gehen **drei didaktische Handlungsbereiche** des Umgangs mit Wissen hervor:

- **Problemorientiertes Gestalten der Lernaufgabe** (zwischen Arbeitsprozeß und den Lernfähigkeiten der Lernenden)
- Die **Wissensvermittlung** von unterschiedlichen Formen des Wissens unter Berücksichtigung der subjektiven Lernhemmungen
- Die **Wissensmoderation** der subjektiven Formen der Aktualisierung und Modifikation von Wissensformen der Lernenden.

Nachfolgend werden wir diese drei Handlungsbereiche näher betrachten, didaktische Prinzipien und didaktisch-methodische Vorschläge zu ihrer Strukturierung

---

lichkeitsentwicklung", so der Titel ihrer Studie, ist u.E. deshalb nur als programmatische Forderung, berufliche Ausbildung zu verbessern, zu verstehen.

benennen, um am Ende unserer Ausführungen anhand eines Beispiels aus unserem Rahmencurriculum »Moderne Software-Entwicklung« eine praktische Umsetzung vorzustellen.

4.1 Problemorientiertes Gestalten der Lernaufgabe

Der Gestaltung der Lernaufgabe kommt eine zentrale Rolle beim didaktischen Umgang mit Wissen zu. Lernaufgaben sind gewissermaßen der Mikrokosmos des Erlernens der beruflichen Handlungskompetenz. In diesem Sinne gilt, daß jede einzelne Lernaufgabe die Funktion der gesamten Ausbildung widerspiegelt, aber auch, daß jede Lernaufgabe in einem bestimmten Verhältnis zu anderen Lernaufgaben steht. Da dieses nicht als Normalität unterstellt werden kann, können wir ein erstes Postulat zur Gestaltung von Lernaufgaben einführen:

1. Lernaufgaben sind im Ausbildungsbezug zu gestalten.

Das Verfügen über berufliche Handlungskompetenz setzt beim Individuum die Integration von unterschiedlichen Einzelkompetenzen voraus. Diese notwendige **Integrationsleistung** wird in der schulischen Lernorganisation des Fächerkanons nicht gefördert, sondern eher behindert. Lernaufgaben im Ausbildungsbezug zu gestalten heißt nun nichts anderes, als daß dieser Zerfaserung der Ausbildungsinhalte zum Teil dadurch entgegengewirkt wird, daß **Lernaufgaben unterschiedlicher Fächer aufeinander bezogen** werden, bzw. **Lernaufgaben fächerübergreifend** an der Gesamtausbildung orientiert werden. Das Gegenteil des Prinzips des Ausbildungsbezugs ist der Fachbezug, beziehungsweise der reine Selbstbezug, in dem die Lernaufgabe gänzlich isoliert betrachtet wird. Die didaktischen Vorteile der Gestaltung von Lernaufgaben im Ausbildungsbezug sind hauptsächlich folgende:

- Die inhaltslogische und didaktische Vernetzung von Lernaufgaben führt zu **Synergieeffekten des Lernens**, die sich in erster Linie in der **Qualität des Lernens** niederschlagen.
- Der Ausbildungsbezug gibt der Lernaufgabe eine sinnvolle Bedeutung, die für den Lernenden nachvollziehbar ist. Der Ausbildungsbezug der Lernaufgabe hat somit auch eine **motivationale Funktion**.
- Erst der Ausbildungsbezug der Lernaufgabe ermöglicht es dem Dozenten, eine **gleichgewichtige Rolle zwischen Wissensvermittler und Wissensmoderator** im konkreten Unterricht wahrzunehmen und damit einen selbsttätigen Wissensumgang (Aktualisierung & Modifikation) auf Seiten der Lernenden zu unterstützen.

Um diese didaktischen Vorteile der Gestaltung von Lernaufgaben im Ausbildungsbezug auch tatsächlich zu realisieren, sind allerdings einige organisatorische Voraussetzungen zu schaffen und einige Regeln (im Sinne des Negationsprinzips unserer didaktischen Konzeption) zu beachten.

Organisatorische Voraussetzungen:

- Der Ausbildungsbezug der unterschiedlichen Lernaufgaben macht nur dann Sinn, wenn zwischen den Dozenten ein **Konsens über das globale Qualifizierungsziel der Ausbildung** und dessen gestalterische Bedeutung besteht. Dieser Konsens kann u.E. zur Zeit nicht vorausgesetzt werden, sondern muß erst in einem Kommunikationsrahmen der Ausbildungsinstitution hergestellt werden.
- Neben diesem globalen Konsens, den wir gewissermaßen als Resultat einer »Metakommunikation« bezeichnen können, ist die **direkte fächerübergreifende Kooperation von Dozenten** notwendig.
- Das starre Zeit- und Raumkorsett der Unterrichtsstunden muß überwindbar sein, wenn es die Gestaltung der Lernaufgabe verlangt (z.B. Projektarbeit).

Einige »Regeln«:

- Wenn Lernaufgaben gestaltet werden, die primär durch die **Integration verschiedener Wissensbereiche** gekennzeichnet sind, sollte die erforderliche **Vermittlung von neuem Wissen minimal** sein, um das situative didaktische Handeln auf die Wissensmoderation konzentrieren zu können. Allgemein gilt, daß die Schwierigkeiten der Lernenden, unterschiedliches Wissen zu integrieren (aktualisieren & modifizieren), größer als vom Dozenten angenommen sind. Dies liegt daran, daß der Dozent einzelne Wissensbereiche analytisch aus einem größeren Zusammenhang gewinnt und somit sich für ihn das Problem der Synthese überhaupt nicht stellt, während die Lernenden zwar abstrakt den größeren Zusammenhang kennen, aber die praktische Integration selbsttätig als eigenständige Aufgabe bewältigen müssen.[102]

---

[102] In der Praxis haben wir es allerdings oft erlebt, daß gerade der analytische Standpunkt des Dozenten häufig durch die »synthetische« Projektarbeit der Lernenden widerlegt wurde. Die Aussage des Dozenten zum Projektanfang, daß die Schüler prinzipiell schon alles wüßten, was sie zur Bewältigung der Projektaufgabe benötigen würden, erwies sich meist dann als falsch, wenn die Schüler einen Lösungsweg einschlugen, der nicht falsch war, aber doch vom antizipierten Weg des Dozenten abwich. Hier stellt sich dann die nur situativ beantwortbare Frage, was bezüglich der Unterstützung selbsttätiger Lernprozesse sinnvoller ist: Den Schülern Lernhilfen zur Verfügung zu stellen, die es ihnen ermöglichen, ihren eigenständigen Lösungsweg

- Wenn Lernaufgaben gestaltet werden, die primär der Vermittlung von neuem Wissen dienen sollen, so ist den Lernenden zuerst der Sinn dieser Lernaufgabe in **Bezug zu einem übergeordneten Ausbildungsziel** zu vermitteln. Das Lernen von **»rules without reasons«** führt zu grundlegend **defizitären Formen des Verstehens.** [vgl. Mellin-Olsen 1981] Dies gilt nicht nur für traditionell hermeneutisch orientierte Wissensbereiche, sondern auch für Fächer wie die Mathematik und das Erlernen des Programmierens. Jedes Wissen muß daher in der **Lernaufgabe als Teil eines subjektiven Erkenntnisprozesses** konzipiert werden. In Analogie zu den Prinzipien der Softwarekonstruktion kann auch die Entwicklung von Lernaufgaben unter den Gesichtspunkten eines **»bottom-up«**- beziehungsweise **»top-down«** - Verfahrens betrachtet werden. Als »Regel« wollen wir **die zyklische Verbindung beider Verfahrensweisen** ansehen. [vgl. Siefkes 1993, S. 103ff.].

Die Gestaltung der Lernaufgabe im Ausbildungsbezug ist eine Grundvoraussetzung erfolgreichen didaktischen Handelns, die immer zu beachten ist. Zielt der Ausbildungsbezug prinzipiell auf die Erhöhung der »Wirksamkeit« von Lernaufgaben im System der Ausbildung selbst, so stellt dieser allerdings noch keineswegs einen gesicherten Realitätsbezug zum tatsächlichen beruflichen Handlungsfeld her. Daher halten wir es für angebracht, ein zweites Prinzip der Gestaltung von Lernaufgaben zu postulieren:

2. Lernaufgaben sind auf den realen Arbeitsprozeß zu beziehen

Wenn die Ausbildung eine einfache Abbildung des realen Arbeitsprozesses wäre, dann wäre dieses Prinzip redundant. Sie ist dies aber weder in ihrer jeweiligen historischen Gestalt, noch kann sie es prinzipiell ihrem Wesen nach sein. Dies haben wir ausführlich bei der Fundierung unserer didaktischen Fragestellung dargestellt: Ausbildung kann sich als organisiertes Lernen nur auf Wissenlernen, soziale Kompetenzen lernen und Identitätsentwicklung beziehen. Die Analyse des Arbeitsprozesses in seine funktionalen Qualifikationsbestandteile, Technikgestaltungskompetenz, Prozeßkompetenz und Kooperationskompetenz liefert dagegen nur das äußere Bezugssystem, auf das die Dimensionen Wissen, soziale Kompetenzen und berufliche Identität in einem **komplexen Anpassungsprozeß** ausgerichtet werden müssen. Lernaufgaben auf den realen Arbeitsprozeß beziehen heißt in »Regeln« gefaßt:

- Wenn arbeitspraktische Erfahrungen, beispielsweise in Form vorgängiger Praktika, bei den Lernenden vorliegen, so hat die **erfahrungsorientierte**

---

weiter zu beschreiten, oder ihnen Lernhilfen anzubieten, die sie wieder auf den vom Dozenten antizipierten Lösungspfad zurückführen?

Gestaltung der Lernaufgaben absolute Priorität. Aber auch die **Simulation von Situationen des realen Arbeitsprozesses** kann als Gestaltungskriterium einer konkreten Lernaufgabe eingesetzt werden (Wichtig und von einschränkender Wirkung ist hier allerdings die »Regel«, daß die erfahrungsorientierte und als Simulation einer realen Arbeitssituation konzipierte Lernaufgabe nur dann sinnvoll ist, wenn sie gleichzeitig Ausbildungsbezüge hat, d.h., wenn die Fokussierung auf ein bestimmtes Wissen, das vermittelt werden soll, möglich ist, bzw. die Bedingung der Wissensmoderation in Form vorhandenen Vorwissens bei den Lernenden gegeben ist).

- Auch wenn die Lernaufgabe als solche nicht unmittelbar auf den Arbeitsprozeß beziehbar ist, so gilt doch, **daß die mit der Lernaufgabe verbundenen Lernziele für den Arbeitsprozeß von Bedeutung sind**. Die Evaluation und Modifikation dieser Lernziele kann letztendlich nur durch eine an den Arbeitsprozeß angelehnte Lernaufgabe erfolgen. Diese bildet gewissermaßen das nächste Ziel innerhalb der Ausbildung (z.B.: Die Vermittlung des Wissens unseres Rahmencurriculums »Moderne Software-Entwicklung« findet in Form der Gestaltung unterschiedlichster Lernaufgaben statt, deren Nahziel eine »Ziel-Lernaufgabe« ist, die die Durchführung eines realitätsnahen praktischen Projekts zum Gegenstand hat).

Zusammenfassend kann gesagt werden: Die Ausrichtung der Lernaufgaben an Situationen des realen Arbeitsprozesses ist immer dann sinnvoll, wenn sich gleichzeitig Ausbildungsbezüge herstellen lassen. Ist dies nicht der Fall, so sollte der Bezug zum realen Arbeitsprozeß in der Gestaltung einer Ziel-Lernaufgabe realisiert werden, die wiederum vorgelagerte Lernaufgaben strukturiert.

Neben dem Ausbildungssystem und dem realen Arbeitsprozeß als äußerlichem Ziel der Ausbildung muß die Gestaltung der Lernaufgabe selbstverständlich die tatsächlichen Lernfähigkeiten der Lernenden berücksichtigen. Diese bilden das dritte Gestaltungspostulat der Lernaufgaben:

3. Lernaufgaben sind auf die Lernfähigkeiten der Lernenden zu beziehen

Eine traditionelle Grundannahme lautet: Die Lernfähigkeit der Lernenden ist prinzipiell im kohärenten Ausbildungsbezug aufgehoben. Kurz: Die subjektiven Lernfähigkeiten werden mit der vorgängigen Ausbildungspraxis in eins gesetzt. Der Didaktiker baut den Ausbildungsgang wie ein Haus auf, er legt zuerst das Fundament, dann mauert er die Grundmauern, um schließlich Stock um Stock in die Höhe zu gelangen. Auch die von uns oben aufgestellte »Regel«, Lernaufgaben im Ausbildungsbezug als aufsteigende subjektive Erkenntnisprozesse zu konzipieren, geht in diese Richtung.

Die praktische didaktische Erfahrung lehrt uns aber, daß die kohärente Konzeption des Ausbildungsbezugs zwar eine notwendige, aber keine hinreichende Bedingung für eine erfolgreiche Gestaltung der Lernaufgabe darstellt.

1. Die realen Lernfähigkeiten der Lernenden sind nicht mit der Beherrschung des jeweils aktuellen Standes der Ausbildung identisch.
2. Die Orientierung der Ausbildung am Qualifizierungsziel Ganzheitliche Arbeitsgestaltung erschwert ein lineares didaktisches Vorgehen, wie es die Metapher des Hausbauens nahelegt. Die Multiperspektivität des Ausbildungsziels bedarf einer multiperspektivischen Ausbildung. Die nur in der aktuellen Situation im Prozeß der Software-Entwicklung bestimmbaren integrativen Leistungen der IT-Fachkräfte müssen als intuitive Erkenntnisleistungen in der Ausbildung didaktisch konzipiert werden. Konkret heißt das: Das **Verstehen eines mathematischen Beweises** aufgrund des Verfügens über bestimmte mathematische Theoreme und Axiome beruht auf einem einmaligen Erlebnis des **Verstehens seiner zwingenden Notwendigkeit**. Ganzheitliche Arbeitsgestaltungskompetenz kann dagegen nicht allein aufgrund vorgängiger Wissensbestände erlangt werden, sondern sie ist das **Produkt der Summe zahlreicher intuitiver Lernerfahrungen**.

Als **intuitive Lernerfahrung** bezeichnen wir das **subjektive Erleben einer Einsicht**, die sich nicht linear zwingend ergibt, sondern die als **Resultat einer subjektiven Integration** unterschiedlicher Wissensfelder im Problemlösehandeln entsteht.

Insbesondere die Gestaltung von »Ziel-Lernaufgaben« muß konzeptionell auf die Ermöglichung intuitiver Lernerfahrungen ausgerichtet sein. Intuitive Lernerfahrungen beruhen in erster Linie allerdings keineswegs auf akkumuliertem Ausbildungswissen, sondern auf der **Verknüpfung von Ausbildungswissen und Alltagswissen**. Die Gestaltung von Lernaufgaben, die das Ziel der Ganzheitlichen Arbeitsgestaltungskompetenz verfolgen, müssen sich daher auch auf Lernfähigkeiten der Lernenden beziehen, die maßgeblich durch Alltagswissen geprägt sind. Als »Regeln« wollen wir daher formulieren:

- Die Gestaltung von Lernaufgaben muß das Aktualisieren von Alltagswissen der Lernenden mit einbeziehen. Sie muß also in irgendeiner Form auf Alltagserfahrungen zurückführbar sein und diese mit Ausbildungswissen verknüpfen.
- Die Gestaltung von Lernaufgaben dient auch der Evaluation von Alltagswissen, das als Lernhemmung der angemessenen Lösung der Lernaufgabe im Weg steht.

- Die Gestaltung der Lernaufgabe schließt die antizipierende Gestaltung umgangssprachlicher Kommunikation über einzelne Problembereiche und Bedeutungen der Lernaufgabe ein. Diese Kommunikation verläuft auf dem schmalen Pfad zwischen zulässigen und nicht zulässigen Analogien, angemessenen und unangemessenen Metaphern.[103]

Zusammenfassend kann u.E. gesagt werden, daß Lernaufgaben nur in dem dreifachen Bezug zur Ausbildung, zum Arbeitsprozeß und zu den Lernfähigkeiten der Lernenden angemessen gestaltet werden können. Jeder dieser Bezüge läßt sich durch einzelne »Regeln« konkretisieren, die allerdings keine fertigen Rezepte darstellen, sondern eher Negationen traditioneller Unterrichtsplanung in dem Sinne sind, daß sie entweder traditionelle Perspektiven kritisieren oder neue Perspektiven einführen.

Während der Bezug der Lernaufgabe zum Arbeitsprozeß und zum Ausbildungskontext mehr oder weniger autonom durch den Kommunikationszusammenhang zwischen den Lehrenden bestimmt wird, dieser Bezug also zum größten Teil in der Planung vorweggenommen werden kann, so ist der Bezug zu den Lernfähigkeiten der Lernenden nur bedingt antizipierbar.[104] Viele Lernhemmungen zeigen sich erst bei der Bearbeitung einer Lernaufgabe durch die Schüler. Da es in den meisten Fällen wenig sinnvoll ist, die Lernaufgabe je nach den Lernhemmungen der Schüler ad hoc umzugestalten, muß der Lehrende über ein differenziertes Instrumentarium von Lernhilfen verfügen, das er je nach aktueller Unterrichtssituation flexibel einsetzt. Dieses Instrumentarium von Lernhilfen bezeichnen wir als Wissensvermittlung und Wissensmoderation.

4.2 Wissensvermittlung und Wissensmoderation

Da wir die Vermittlung und Moderation von Wissen als ein gemeinsames didaktisches Handlungsfeld ansehen, scheint uns eine getrennte Betrachtung der beiden Begriffe hier wenig sinnvoll. Jede Vermittlung von Wissen verweist u.E. notwendig auf eine Moderation von Wissen, sowie jede Form der Moderation in strengem Sinne auch eine Vermittlung von Wissen darstellt. Trotzdem hat die

---

[103] Auch hier gilt das Postulat Ottes, daß eine gute Fachdidaktik sich letztendlich immer auch auf wissenschaftsphilosophische Fragestellungen beziehen muß. Die Genese von Erkenntnis (Didaktik) kann nicht von der wissenschaftsphilosophischen Erkenntnistheorie getrennt werden [vgl. Otte 1994, S. 7].

[104] Das bedeutet freilich nicht, daß auf die Antizipation von Lernfähigkeiten verzichtet werden könnte. Gerade eine gründliche Auseinandersetzung mit notwendigen und angenommenen Lernfähigkeiten in der Planung ermöglicht es dem Lehrenden, tatsächliche, nicht antizipierte Lernhemmungen zu erkennen.

Differenz von Wissensvermittlung und Wissensmoderation einen für das praktische didaktische Handeln relevanten Sinn, der sich in der unterschiedlichen Ausgangsposition des didaktischen Handelns festmachen läßt:

- Die Wissensvermittlung geht davon aus, daß die Lernhemmungen des Lernenden durch das Fehlen eines bestimmten Wissens verursacht sind. Dieses fehlende Wissen ist kein subjektives Konstrukt des Lernenden, sondern ist eindeutig sachlogisch fundiert.
- Die Wissensmoderation geht davon aus, daß dem Lernenden prinzipiell das zur Lösung der Lernaufgabe notwendige Wissen zur Verfügung steht. Nicht fehlendes Wissen verhindert eine angemessene Problemlösung, sondern ein **subjektives Konstrukt des Wissens**, das Aktualisierungen blockiert und spezifische Modifikationen des Wissens unwahrscheinlich macht.

Im folgenden legen wir einige Aspekte der Wissensvermittlung dar und zeigen die Implikationen auf, die von diesen Aspekten auf die Notwendigkeit und Art und Weise der Wissensmoderation ausgehen. Daran anschließend wählen wir das umgekehrte Verfahren, indem wir von Problemen der Wissensmoderation ausgehen und auf ihre Bedeutung für die Wissensvermittlung hinweisen.

4.2.1 Wissensvermittlung: Zwischen Aneignung und Beschleunigungsdidaktik

Wissensvermittlung befindet sich generell im Zielkonflikt, entweder eine möglichst große Menge Wissen möglichst schnell zu vermitteln, oder eine weniger große Stoffmenge von den Lernenden selbsttätig erarbeiten zu lassen. Dieser **Widerspruch von Quantität und Qualität** ist auch durch die besten didaktischen Methoden nicht kompensierbar, da über sie nicht im didaktischen Handeln entschieden wird, sondern in den innerpsychischen Lernprozessen der Lernenden. Die Fähigkeit, adaptives Wissen zu reproduzieren, erscheint auf den ersten Blick wesentlich effizienter als der Prozeß mühevoller subjektiver Aneignung von Wissen. Die Argumente, die in der Regel für den zweiten, mühevolleren Weg vorgebracht werden, sind bekannt. Wir wollen sie hier trotzdem kurz skizzieren, weil sie für die Umsetzung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« von besonderer Bedeutung sind:

- Wissen, das den Aspekt des selbsttätigen Erarbeitens besitzt, hat eine längere Verfallszeit (**Aktualisierungsaspekt des Wissens**).
- Die erfolgreiche selbsttätige Aneignung von Wissen fördert die Motivation, sich aktiv beruflichen Problemen zu stellen (**Motivationsaspekt des Wissens**).

- Aktiv angeeignetes Wissen wird eher mit anderem Wissen sowohl horizontal als auch vertikal verknüpft, da es schon ursächlich über Verknüpfungen angeeignet worden ist (**Modifikationsaspekt des Wissens**).
- Der Wert einer reinen Wissensreproduktion wird für die Handlungskompetenz zunehmend geringer. Nicht das mentale Verfügen über reproduzierbare Informationen, sondern das Wissen über Zugänge zu mediengespeicherten Informationen wird wichtiger (**medialer Speicheraspekt adaptiven Wissens**).

Die Vorteile des selbsttätigen Aneignens von Wissen durch die Lernenden haben zur Entwicklung zahlreicher **handlungsorientierter didaktischer Methoden** geführt. Am bekanntesten sind hier die **Projektmethode** und die **Leittextmethode** [vgl. Sell/ Fuchs-Frohnhofen 1993, S. 132ff.]. Trotz der zahlreichen und erfolgreichen Erfahrungen mit diesen Methoden kann kaum von ihrer breiten Durchsetzung in der Praxis gesprochen werden.

Mit dem Ziel, dieses »Praxisloch« zu überwinden, hat sich eine didaktisch-methodische Forschung ausdifferenziert, die unter dem **Paradigma der Steigerung der Problemlösefähigkeit** zahlreiche Einzelmethoden als systematische Lernhilfen entwickelt hat. Da diese Einzelmethoden in einen systematischen Methodenverbund eingeordnet werden, der anscheinend in idealer Weise allen unseren Anforderungen bezüglich der Umsetzung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« entspricht, wollen wir hier kurz auf den Nutzen dieser Konzepte eingehen. Immerhin versprechen sie nicht weniger, als daß auch die Wissensvermittlung als Wissensaneignung methodisch-systematisch steuerbar ist.

Ausgehend von Ablaufdiagrammen, die analytisches, synthetisches und dialektisches Problemlösen systematisch erfassen, werden Problemlösetechniken und -strategien entwickelt, die zusammen mit sozialen Kompetenzen eine umfassende **Problemlösekompetenz** ergeben [vgl. Sell/ Fuchs-Frohnhofen 1993, S. 132ff.]. Neben der Identifizierung des »problematischen« Sachverhalts und der Bestimmung des Problemtyps tritt das systematische Abarbeiten seiner Lösung: Auswählen der Überwindungsstrategie, Auswählen der Überwindungstaktik, Auswählen der Überwindungsmethode. Der Lehrer hat hier in erster Linie die Funktion eines Moderators beziehungsweise Lernberaters. Selbst wenn er Wissen vermittelt, so tut er dies immer schon im Hinblick auf handlungsrelevantes Problemlösen. Die didaktisch-methodisch orientierte Erforschung der Problemlösefähigkeit erfaßt nicht nur das Problemlösen selbst als letztendlich methodisch beherrschbares Problem, sondern liefert auch dem Lehrer die entsprechenden

Moderationsmethoden und macht konkrete Vorschläge medialisierter Wissensvermittlung [vgl. Sell/ Fuchs-Frohnhofen 1993; Höpfner 1991].

Ohne daß wir hier auf diese Methoden und ihre Begründungen im einzelnen eingehen können, erscheint es uns an dieser Stelle doch notwendig, eine prinzipielle Kritik an diesem methodenfixierten Vorgehen zu formulieren. Grundsätzliche Probleme des pädagogischen Verhältnisses und der Fachdidaktik werden ebenso wie das Lösen dialektischer Probleme simplifiziert, um sie methodisch in den Griff zu bekommen. Am Schluß steht ein **Methodenfetischismus,** der die Wahrnehmung des didaktischen Handelns im pädagogischen Verhältnis **instrumentell verkürzt** und damit verzerrt. Dieser Methodenzwang wird spätestens bei der »notwendigen« Instrumentalisierung von Kreativität deutlich, wie sie z.B. Sell vorschlägt: Ungeachtet gegensätzlicher Ergebnisse der empirischen Kreativitätsforschung hält Sell an dem kreativitätsfördernden Effekt des Brainstormings und anderer Kreativitätstechniken fest und bezieht sich gleichzeitig auf eine Theorie der Kreativität als "laterales Denken" [de Bono 1971], die zwar weltweit propagiert wird, aber destotrotz keinerlei wissenschaftliche Bestätigung erfuhr [vgl. Heller 1992]. Wir schließen uns der Kritik Hellers an, der resümiert: "Die Attraktivität reduktionistischer Modelle oder »Expreßkurse« zur Denkschulung ist aber nicht nur in der Praxis, sondern auch in der Wissenschaft ungebrochen, was man vielleicht als Ausdruck unserer Sehnsucht nach der »einfachen« Wahrheit oder - profaner - menschlicher Bequemlichkeit interpretieren könnte" [Heller 1992].

Die Entscheidung für eine subjektorientierte Didaktik, die auf die selbsttätige Aneignung von Wissen setzt, kann u.E. nicht auf ein Methodenkorsett zurückgreifen, das weder den Lerninhalten noch der Unterstützung autonomer Lernprozesse gerecht wird.

Trotz dieser grundsätzlichen Kritik halten wir viele dieser Methoden für geeignete Lernhilfen[105], um auf bestimmte Lernhemmungen zu reagieren. Insbesondere stellen sie unterschiedliche Formen der Wissensmoderation[106] dar, die sich direkt auf die Vermittlung von Wissen beziehen. Nur das selbstgesetzte Ziel, die systematische Gestaltung von Unterricht mit der Vorgabe, eine allgemeine Problemlösekompetenz herzustellen, können diese Methoden nicht einlösen.

---

[105] So z.B.: Moderationshinweise für die Moderation einer themenzentrierten Interaktion als bewußte Herstellung der Balance von Selbstorientierung, Interaktionsorientierung und Aufgabenorientierung [Sell/ Fuchs-Frohnhofen 1993, S. 124ff.]

[106] So z.B.: Visualisierung, Fragetechnik, Gedankenprotokoll und Analogienbildung [Sell/ Fuchs-Frohnhofen 1993, S. 57]

4.2.2 Wissensvermittlung: Zwischen Medialisierung und direkter Kommunikation

Wer sich mit manchen modernen didaktischen Methoden beschäftigt, mag kaum noch glauben, daß Wissen in direkter sprachlicher Kommunikation vermittelt werden kann. Multimediale Trainer trachten danach, alle Sinnesorgane didaktisch zu verdrahten, ängstlich darum bemüht, jede Sinnesstörung auszuschließen. Gleichzeitig wird allenthalben ein Nachlassen der Konzentrationsfähigkeit beklagt, die dann wiederum durch größeren Medieneinsatz kompensiert werden soll.

Eine Didaktik, die sich die Vermittlung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« zum Ziel gesetzt hat, sollte dagegen nach dem Prinzip vorgehen: So viel direkte sprachliche Kommunikation wie möglich und so viele didaktische Medien wie unbedingt nötig. Dies hat folgende Gründe:

- Direkte sprachliche Kommunikation ist das flexibelste didaktische Medium. Nur sprachliche Kommunikation kann situativ auf die subjektiven Lernbedürfnisse der Lernenden eingehen.
- Direkte sprachliche Kommunikation ist das einzige Medium, mit dem die unterschiedlichen Geltungsansprüche von Kommunikation transportiert werden können.
- Komplexe Begriffe sind nur sprachlich erklärbar. Visualisierungen dienen nur der Illustration.
- Sprachkompetenz ist selbst eine »subjektive« Bedingung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz«.
- Direkte sprachliche Kommunikation ist das einzige Medium, das sich auf alle Wissensformen beziehen und eine Balance zwischen Wissensvermittlung und Wissensmoderation herstellen kann.
- Nur direkte sprachliche Kommunikation ermöglicht eine dialogische Form des Unterrichts, die Grundvoraussetzung für ein pädagogisches Verhältnis und damit auch für die Transparenz des Lernprozesses ist.

Medieneinsatz erscheint uns nur dann didaktisch sinnvoll, wenn er sprachliche Kommunikation unterstützt, ohne sie zu verzerren. So ist es durchaus angebracht, komplexe Modelle graphisch zu visualisieren, weil sie so leichter aktualisierbar und überschaubar sind. Allerdings ist es fatal, Verstehen über graphische Modelle herstellen zu wollen, da dieses prinzipiell an Sprache orientiert ist.

Eine ebenso fragwürdig wie verfehlte Nutzung der Visualisierung ist es, sprachliche Kommunikation generell durch Medieneinsatz strukturieren zu wollen.[107] Die Aufmerksamkeit und Konzentrationsfähigkeit wird auf die jeweils aktuelle Form der Kommunikationsrepräsentanz gelenkt. Die Vielschichtigkeit der Kommunikationsbezüge wird nur scheinbar erhöht. Massiver Medieneinsatz hat allerdings oftmals den Charakter einer selffulfilling prophecy: Ohne Pinnwand, Tageslichtprojektor und Flip-Charts in einer Gruppe themenzentriert zu kommunizieren, ist für manche Teilnehmergruppen unvorstellbar.

Im folgenden wollen wir einige Kriterien für den sinnvollen Medieneinsatz vorschlagen:

- **Illustrative Ergänzung**: Medien illustrieren sprachlich kommuniziertes Wissen. In der Regel erfolgt die Illustration nach der sprachlichen Vermittlung beziehungsweise begleitet die sprachliche Wiederholung von Wissenseinheiten.
- **Visualisierte Gedächtnisstütze**: Visualisierungen eines Modells können als Gedächtnisstütze bei der sprachlich-begrifflichen Rekonstruktion von Modellen hilfreich sein.
- **Strukturierungshilfen**: Pinnwände etc. können bei der Strukturierung der Form-Rekonstruktion von Kommunikationsprozessen ex post eingesetzt werden.
- **Videoaufnahmen** ermöglichen eine **Selbstkonfrontation**, die ohne dieses Medium unmöglich wäre. Insbesondere für die Vermittlung und Moderation von sozialem Wissen besitzt dieses Medium eine hervorgehobene Rolle.
- **Medien als Informationsträger**: Medien, wie Filme und Textdokumente, eignen sich als Träger authentischer Informationen aus anderen Realitätsbereichen (Wissenschaft, Arbeitswelt). Als solche bieten sie Kommunikationseinstiege.

Sprachliche, direkte Kommunikation bleibt, trotz der sequentiellen didaktischen Nützlichkeit anderer Medien, das einzige Medium, in dem sich eine subjektorientierte Didaktik als Einheit von Wissensvermittlung und Wissensmoderation

---

[107] Das Klagen über die schlechte Medienausstattung findet sich auch bei Sell: "Nur bezüglich der Medien hat sich noch nicht durchgängig durchgesetzt, daß neben Tageslichtschreibern und Flip-Charts auch entsprechende Präsentationsflächen wie Pinnwände oder Wandschienen ausreichend zur Verfügung gestellt werden. Das Mitbringen des eigenen Moderatorenkoffers mit Utensilien wie bunte Kärtchen, Nadeln, Klebeband, Filz- und Folienschreiber, etc. ist schon zur Selbstverständlichkeit geworden" [Sell/ Fuchs-Frohnhofen 1993, S. 107]

entwickeln kann. Mangelnde Mediennutzung ist das seltenste Defizit bei der beruflichen Handlungskompetenz der Lehrenden, unangemessenes kommunikatives Handeln das häufigste.

4.2.3 Wissensvermittlung: Zwischen den Formen des Wissens

Verschiedene Formen von Wissen stellen nicht nur einen unterschiedlichen Zugang zur Wirklichkeit dar, sondern verlangen auch unterschiedliche Formen der Aneignung. Wir differenzieren im folgenden zwischen Wissensformen, die sowohl den **Zugang zur Wirklichkeit** gegenstandsbezogen unterschiedlich strukturieren, als auch eine unterschiedliche **Form der Aneignung** auf Seiten des Subjekts erfordern, ohne allerdings den Anspruch zu erheben, damit alle Formen des Wissens erfaßt zu haben.[108]

- **Nominalistisches Wissen**: Als nominalistisches Wissen bezeichnen wir Wissen, das einzelne Gegenstände bezeichnet, bzw. einzelnen Handlungen eine Bezeichnung zuordnet. Ebenso bezeichnen wir beispielsweise auch die Befehlssyntax eines Betriebssystems als nominalistisches Wissen, wenn sie unabhängig von der spezifischen Anwendung angeeignet wird. (Die Anwendungsfunktionen von Betriebssystemen sind bereits bekannt, während die spezifische Befehlssyntax eines Betriebssystems erlernt wird.)
- **Anwendungswissen**: Als reines Anwendungswissen bezeichnen wir algorithmisiertes Wissen, das dem Aufbau einer Handlungssequenz dient. Anwendungswissen ist die systematische, auf ein Handlungsziel ausgerichtete Verknüpfung von nominalistischem Wissen. Die Form der Verknüpfung stellt die Algorithmisierung dar (z.B. systematisches Programmieren nach formalen Regeln). In reiner Form liegt Anwendungswissen allerdings ebensowenig wie die anderen Wissensformen vor.
- **nomologisches Erschließungswissen**: Nomologisches Erschließungswissen hat wie das algorithmisierte Wissen eine Wenn-Dann-Struktur. Diese Struktur ist aber nicht jeweils willkürlich gesetzt, sondern ergibt sich aus wenigen grundlegenden axiomatischen Setzungen. Als wichtigstes nomologisches Erschließungswissen gelten die Mathematik und die Logik. Da die Axiome der Mathematik - entgegen einem weitverbreiteten Mythos [vgl. Otte

---

[108] Wie wir bei der Erörterung des didaktischen Umgangs mit der Identitätsproblematik sehen werden, kann es durchaus sinnvoll sein, einen anderen Ausgangspunkt und andere Kriterien zur Klassifizierung des Wissens zu wählen. So können wir beispielsweise nach der Form der emotionalen und motivationalen Bedeutung des Wissens unterscheiden (ebenso nach der sozialen Orientierungsfunktion, der sozialen Genese und der integrierenden Funktion von Wissen für das einzelne Subjekt).

1994] - nicht frei von der Realitätserfahrung sind, kann dieses strenge Wissen auch Realität erschließen. Nomologisches Erschließungswissen strukturiert in der Regel algorithmisches Anwendungswissen.

- **heuristisches Erschließungswissen**: Heuristisches Erschließungswissen bezieht sich auf Modelle und Begriffe, die zum Ausgangspunkt von Analogien und Metaphern werden, die die Handlungsorientierung prägen (z.B. liegt ein Projektmodell nur als heuristisches Erschließungswissen vor). Ebenso setzt das nomologische Erschließungswissen oftmals heuristisches Wissen voraus (z.B. können formale Methoden als nomologisches Erschließungswissen fungieren, wenn ihre Brauchbarkeit durch heuristisches Erschließungswissen assoziiert wird). Auch das heuristische Wissen hat einen nomologischen Kern, der allerdings nicht die Form eines deduktiv-nomologischen Syllogismus hat, sondern als induktiv-empirischer Syllogismus zu verstehen ist.
- **Reflexionswissen**: Als Reflexionswissen bezeichnen wir normatives und theoretisches Wissen, das eine distanzierte Beurteilung von Handlungen und Gegenständen zuläßt. Reflexionswissen liegt immer dann vor, wenn Bezüge jenseits des eigentlichen Handlungsziels und des eigentlichen Handlungsgegenstandes hergestellt werden (z.B. werden Handlungen bezüglich eines übergeordneten Wertesystems reflektiert und nicht intendierte Nebenfolgen von Handlungen betrachtet). Reflexionswissen dient weniger der Lösung eines vorhandenen Problems als der Auffindung neuer Problemfelder.

Die didaktische Umsetzung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« erfordert ein »**harmonisches Wissensmodell«**, in dem alle obengenannten Wissensformen gleichermaßen durch didaktisches Handeln vermittelt und moderiert werden. Für die Wissensvermittlung bedeutet dies:

Bei der Vermittlung einer bestimmten Wissensform sind die Beziehungen zu anderen Wissensformen zu beachten und zum gegebenen Zeitpunkt (beispielsweise nach Abschluß einer Lerneinheit oder bei der Durchführung eines praktischen Projekts) zu problematisieren. Als Grundproblematik der Ganzheitlichen Arbeitsgestaltung bietet das **Verhältnis von softwaretechnischer Formalisierung und lebendiger Arbeit** zahlreiche inhaltliche Bezüge zu den unterschiedlichen Wissensformen. Formale Methoden sind z.B. nicht hinreichend als algorithmisiertes Anwendungswissen bestimmbar. Sie haben vielmehr eine Nominalstruktur, einen nomologisch-deduktiven Hintergrund (und selbst wiederum nomologisch-deduktive Funktion), einen induktiv-empirischen Bezug

zur spezifischen Projektrealität und können, da sie nichtintendierte Nebenfolgen auslösen können, zum Gegenstand von Reflexionswissen werden.

Da das Verhältnis von softwaretechnischer Formalisierung und lebendiger Arbeit in der traditionellen IT-Ausbildung einseitig zugunsten algorithmisierbaren Anwendungs- und Erschließungswissens aufgelöst wird, kann als programmatische didaktische Handlungsorientierung die Forderung **»Vom algorithmisierten zum heuristischen Wissen!«** aufgestellt werden.

Die Vermittlung eines »harmonischen Wissensmodells« macht die Wissensmoderation als zentralen Bestandteil des didaktischen Handelns zwingend notwendig. Im folgenden werden wir einige Aspekte der Wissensmoderation und ihren Bezug zur Wissensvermittlung näher betrachten.

4.2.4 Wissensmoderation: Zwischen nomologischer Perspektive und Kreativität

Kreativität ist eine personale Grundvoraussetzung für die Software-Entwicklung. Dieser Aussage werden fast alle IT-Fachkräfte zustimmen und jeweils etwas anderes damit meinen. Kreativität ist ein äußerst diffuser Begriff. Wenn er zum Bezugspunkt des didaktischen Handelns werden soll, muß er daher vorab geklärt werden. Diese Klärung kann dadurch erfolgen, daß wir uns zwei gängige - durchaus forschungsrelevante - Extrempositionen des Verständnisses von Kreativität vor Augen führen: Die erste Position assoziiert Kreativität mit dem Mythos des Genies. Da Genies kaum didaktisch herstellbar sind, ergibt sich aus dieser Position keine didaktische Handlungsorientierung. Die zweite Position behauptet dagegen, daß jeder Mensch kreativ sei. Didaktisches Handeln hat demnach nur die Aufgabe, den Menschen bei der Entfaltung seiner Kreativität durch Kreativitätstechniken zu unterstützen. Wir wählen im folgenden eine mittlere Position, die dadurch zustande kommt, daß wir Kreativität im Rahmen der sogenannten "4P-U-Interaktion" [Urban 1993] betrachten. Die "4P-U-Interaktion" steht für die Interaktion von "Person-Problem-Prozeß-Produkt im Rahmen der Mikro- und Makro-Umwelt" [ebd.]. Für die Software-Entwicklung ergeben sich damit - stark vereinfacht - folgende unterschiedliche Perspektiven:

- Die **kreative Wahrnehmung von Problemen** durch die Person und die Vertretung dieser Problemsicht in der **Mikroumwelt** (z.B. im Team). Mit der kreativen Problemsicht wird zwar gleichzeitig ein bestimmter Prozeß und ein bestimmtes Produkt assoziiert, aber keineswegs determiniert.
- Die **kreative Gestaltung des Entwicklungsprozesses** durch die Person. Der kreative Prozeß steht gewissermaßen im Mittelpunkt der Kreativität, da

in ihm sowohl neue Problemsichten entstehen, als auch die kreative Gestaltung des Produkts Form annimmt.

- Die **Kreativität des Produkts** als vergegenständlichte Kreativität der Person ist nur insofern kreativ, als dieses Produkt "Sinn und Bedeutung in sich trägt und eine bedeutsame und angemessene Lösung für ein gegebenes, gefundenes, sensitiv wahrgenommenes Problem darstellt" [Urban 1993]. Die Kreativität des Produkts ist von der **Makro-Umwelt** abhängig, die die Originalität und Elaboration des Produkts letztendlich beurteilt.

Diese Dreiteilung der Kreativität in interaktive Dimensionen und Umweltbezüge, in kreative Problemsicht, kreatives Prozeßhandeln und kreative Vergegenständlichung führt zu einem differenzierten Begriff von den personalen Fähigkeiten, die die Grundlage für vollständiges kreatives Handeln bilden: "Eine breite Wahrnehmung und ein tief strukturiertes, vieldimensionales Netzwerk an Wissen/ Informationen ist Voraussetzung für divergentes und assoziatives Denken, das wiederum begleitet und unterstützt wird von konvergenten Formen des Denkens" [ebd.]. Darüber hinaus lassen sich aus dieser Dreiteilung auch didaktische Regeln ableiten, die kreatives Handeln ermöglichen, ohne sich in assoziativem Denken zu verlieren:

- Bei der Vermittlung von nomologischem Wissen und seiner Anwendung ist strengstens auf den Geltungsbereich des nomologischen Wissens zu achten. Nomologisches Prozeßwissen führt oftmals zu einer eingegrenzten Problemsicht und einem verengten Gestaltungsspielraum. Nomologisches Wissen wird als Resultat von Entscheidungswissen eingesetzt, sollte dieses aber nicht determinieren. Die didaktische Methode besteht in einer kritischen Fragetechnik. Gleichzeitig sollte der Lehrende alternative Sichtweisen einführen, um deutlich zu machen, daß in Systementwicklungsprozessen vom Anfang an Entscheidungen eine große Rolle spielen.
- Die analytische Dreiteilung des kreativen Handelns sollte didaktisch genutzt werden, indem beispielsweise Unterrichtsphasen gestaltet werden, die sich ausschließlich der Multiperspektivität möglicher Problemsichten widmen, ohne diese sofort auf Prozeßtechniken und Produktvorgaben zu beziehen (Genauso können auch unterschiedliche Prozeßmethoden entwickelt werden, die vom gleichen Problem ausgehen und zu gleichen Produkten führen).
- Das kreative Handeln in Prozessen hat Auswirkungen auf die Problemsicht und auf die Lösungsgestalt des späteren Produkts, die oftmals nicht reflektiert werden. Die Interaktionszusammenhänge von Problemen, Prozessen und Produkten sind den Lernenden durch Nachfragen bewußt zu machen

(Wenn wir ein spezifisches Verfahren wählen, welches Problem versuchen wir damit zu lösen? Als welches Problem prozessieren wir unser Problem, wenn wir es so behandeln? Ist diese Problemsicht adäquat? Was berücksichtigen wir nicht? u.ä.).

- Eine in der Ausbildung von IT-Fachkräften viel zuwenig angewandte didaktische Methode besteht in der Kritik fertiger Softwareprodukte. Besonders wertvoll ist es u.E., den Prozeß der Software-Entwicklung von einem fremden Endprodukt aus zu rekonstruieren (Welcher Entwicklungsprozeß mit welchen Entwicklungsmethoden hat wohl stattgefunden? Welches ursprüngliche Problem wurde wohl bearbeitet? Welche Alternativen hätte es vielleicht gegeben?). Die Rekonstruktion ist nicht nur erkenntnistheoretisch, sondern auch didaktisch die evidenteste Methode, die Annahme zu zerstören, daß Systementwicklung nomologisch nach dem Prinzip des »one best way« ablaufen würde. Gleichzeitig wird kreatives Denken fast »zwanghaft« gefördert.

Die Moderation des Wissens, die sich die Erhöhung der Kreativität der Lernenden in der Software-Entwicklung zum Ziel setzt, kommt freilich nicht ohne die Vermittlung von neuem Wissen aus. Schon allein die von Urban angesprochene Notwendigkeit eines »vieldimensionalen Netzwerks an Wissen« macht dies deutlich:

»Vieldimensionales, vernetztes Denken« verspricht als modisches Schlagwort eine Identität des Denkenden mit dem Komplexen, die es so nicht gibt: Wir können nicht vieldimensional und vernetzt denken, sondern immer nur einzelne Verknüpfungen herstellen. **Vieldimensionalität und Vernetztheit sind Resultate dieser Verknüpfungen.** Wer das Netz aber knüpft, kann das Netz nicht sehen, weiß also nicht, ob die Menge der Wissensverbindungen, die er herstellt, ein Netz ergeben, das dem Gegenstand gerecht wird. Um sich nicht ins Unendliche zu verstricken, braucht der Handelnde ein als Äußeres gesetztes, **hierarchisch strukturiertes Orientierungswissen**. Dieses Orientierungswissen erstreckt sich sowohl auf die Entwicklung einer konkreten **Zielorientierung** als auch auf **Zusammenhänge relevanter Dimensionen**. Für die Entwicklung von Software haben wir dieses Orientierungs- und Zusammenhangswissen im Rahmencurriculum »Software-Entwicklung« ausgearbeitet. Die Vermittlung von diesem Orientierungs- und Zusammenhangswissen ist der strukturierende Hintergrund der Wissensmoderation, die auf Multiperspektivität und Kreativität abzielt.

Gleichzeitig verweist die Vieldimensionalität des vernetzten Denkens auch darauf, daß zur kreativen Lösung von Fachproblemen keineswegs nur Fachwissen, fachliches Orientierungs- und Zusammenhangswissen benutzt werden kann,

sondern ständig auch sogenanntes Alltagswissen modifiziert und angewendet werden muß. Während das als Äußeres gesetzte Orientierungs- und Zusammenhangswissen gewissermaßen sicherstellt, daß die einzelnen Verknüpfungen auch einen »objektiven« Sinn erhalten, also ein Netz ergeben und kein Wirrwarr von Knoten, stellt das Alltagswissen die subjektive Basis der Verknüpfungen dar.[109]

4.2.5 Wissensmoderation: Zwischen Analogien und Metaphern - Grenzen und Möglichkeiten der Wissensmoderation von Alltagserfahrungen

Die didaktische Wissensmoderation, die Orientierungen und Zusammenhänge zwischen unterschiedlichen Wissensbezügen herstellen will, verwendet als zentrale Mittel Analogien und Metaphern. Auch hier wird wieder die strukturelle Verwandtschaft von Fachdidaktik und Wissenschaftsphilosophie deutlich:

- Der Gebrauch von Analogien kann sich auf die Strukturverwandtschaft einzelner Handlungssituationen und -gegenstände beziehen, das heißt, auf allgemeine Begriffe rekurrieren. Solche Begriffe sind z.B. ein allgemeiner Arbeitsbegriff, der Werkzeugbegriff, der Maschinenbegriff, der Systembegriff, der Projektbegriff, der Form-Prozeßbegriff. Die intellektuelle Leistung der Wissensmoderation besteht dann darin, spezifische Problemsichten, Prozesse und Produkte der Software-Entwicklung auf allgemeine Begriffe zu beziehen, die sich als Alltagswissen reformulieren lassen. Allgemeine Begriffe bilden den »logischen« Kern dieser Analogien. Das heißt aber auch: Angemessene Analogien setzen eine Kritik allgemeiner Begriffe voraus (z.B.: Oftmals werden Softwaresysteme begrifflich als Arbeitswerkzeuge beschrieben, um auf den Zusammenhang von Technik- und Arbeitsgestaltung hinzuweisen. Die Reformulierung des Werkzeugbegriffs findet sich dann beispielsweise in der alltagsbezogenen Analogie des Werkzeugs als Hammer wieder. Die Gefahr eines unkritischen Analogieverständnisses besteht nun aber, wenn kein kritischer Werkzeugbegriff vorliegt, der das Werkzeug-Sein näher bestimmt[110]).

---

[109] Diese Fundierung objektivierter Verknüpfungen in der subjektiven Basis des Alltagswissens und der Alltagserfahrungen ist keineswegs nur ein didaktisches Vehikel, um komplexe Sachverhalte zu verdeutlichen: Selbst hochabstrakte Theorien, wie z.B. die Quantenphysik oder die Systemtheorie Luhmanns, müssen den Bezug zur alltagsbezogenen Umgangssprache herstellen, um ihre Entwicklungsfähigkeit zu erhalten [vgl. zum Konzept der Komplementarität in der Quantenphysik: Otte 1994, S. 275; zur Systemtheorie: Luhmann 1984, S. 13].

[110] Als alltagsbezogene Analogie findet sich permanent der Hammer. Dies liegt insbesondere auch an der fragwürdigen Rezeption Heideggers durch Informatiker

- Die Analogie kann sich auf die Strukturverwandtschaft der phänomenologischen Erscheinung und ihrer Deutung beziehen. Wir nennen diese nicht auf allgemeine Begriffe rekurrierenden Analogien Metaphern. Die Gültigkeit von Metaphern besteht in der nicht weiter begründbaren Ähnlichkeit von Erscheinungen unterschiedlicher Kontexte. Gleichzeitig ermöglichen Metaphern Assoziationen, die durchaus zu neuen Problemsichten und Erkenntnissen führen können. Dies liegt daran, daß Metaphern per se einen Perspektive-Wechsel erfordern. (Wenn wir z.B. behaupten, daß erfolgreiche Projektarbeit in erster Linie nicht das Resultat der Leistungen einzelner Projektmitglieder ist, sondern maßgeblich durch die Kooperation der einzelnen Mitglieder bestimmt ist, so können wir unsere Behauptung mit der Metapher einer Fußballmannschaft unterstützen. Wenn die Metapher akzeptiert wird, können z.B. Erfolgskriterien, die für ein Fußballteam gelten, hinsichtlich ihrer Bedeutung für ein Projektteam betrachtet werden. Gleichzeitig kann die Frage aufgeworfen werden, welche Faktoren die jeweiligen Teamarbeiten unterscheiden usw.).

  Als praktisches Prinzip des didaktischen Gebrauchs von Metaphern gilt darüber hinaus, daß der motivationale Aspekt der emotionalen Besetzung von Metaphern beachtet wird.

Analogien und Metaphern bilden zweifelsohne Grundpfeiler der Wissensmoderation. Wenn ihre erkenntnistheoretischen und didaktischen Grenzen beachtet werden, sollten sie häufig als Lernhilfen herangezogen werden.

Die bisher dargestellten Formen der Wissensmoderation haben sich in erster Linie auf die Hinführung zu einem bestimmten Wissen konzentriert. Tatsächlich steht die didaktische Wissensmoderation aber oftmals vor dem Problem, die Lernenden auf einen Irrtum aufmerksam zu machen, das heißt, sie aus einer »Sackgasse« herauszumoderieren. Dieser Aspekt der Wissensmoderation soll den Abschluß unserer Erörterung des didaktischen Umgangs mit Wissen bilden.

[Züllighoven 1992; Winograd/ Flores 1989]. Der Mathematiker und Wissenschaftsphilosoph Serres fragt sich dagegen u.E. zurecht: "Wie wird diese Philosophie des Hackbeils, ich sage erst gar nicht einem Computer, sondern allein schon einer einfachen Maschine gerecht?" [Serres 1987, S. 100f.]. Vgl. auch kritisch: [Bardmann/ Dollhausen 1992].

4.2.6 Wissensmoderation: »Moderation aus der Sackgasse« oder der Irrtum[111] zwischen Zufall und System

Der Lehrende wird immer wieder damit konfrontiert, daß die »subjektive« Wirklichkeit der Lernenden nicht der »objektiven« Wirklichkeit entspricht, ohne daß dies den Lernenden bewußt ist. Diese Diskrepanz zwischen den Wirklichkeiten kann in einem einfachen Irrtum bestehen, sie kann aber auch eine Sackgasse sein, die vom Lernenden immer wieder systematisch gegangen wird. Wir wollen hier drei Formen des Irrtums idealtypisch unterscheiden, da sie jeweils andere Formen der didaktischen Wissensmoderation erfordern:

- **Der zufällige Irrtum**: Die Lernaufgabe wird vom Lernenden falsch gelöst, weil er z.B. einen Parameter übersehen hat.
- **Der lernförderliche systematische Irrtum**: Ein systematischer Irrtum wirkt dann lernförderlich, wenn er im Bearbeitungsprozeß der Lernaufgabe als solcher evident wird.
- **Der systematische Irrtum als Sackgasse**: Der Irrtum endet in einer Sackgasse, die dem Lernenden keinen Ansatzpunkt bietet, den Irrtum zu erkennen, beziehungsweise, der systematische Irrtum führt zu falschen Lösungen, die nicht als solche für den Lernenden evident sind.

Diesen drei Irrtumstypen können folgende didaktische Formen der Wissensmoderation zugeordnet werden:

- Der zufällige Irrtum erfordert die **Fokussierung der Wahrnehmung des Lernenden** auf den Irrtumsgrund. Die Problematik des didaktischen Handelns liegt hier eher in der Gefahr einer falschen Analyse des Irrtumstyps als in dessen Behebung.
- Der lernförderliche systematische Irrtum führt unausweichlich in eine Situation, in der er evident wird. Die Wissensmoderation konzentriert sich auf diese **Lernsituation des Irrtums**, ohne vorher korrigierend einzugreifen. Trotz des positiven lernpsychologischen Effekts, den diese Formen von Irrtümern haben, sollten sie nicht absichtsvoll eingesetzt werden: Die Lernenden bewußt auf einen in die Irre gehenden Weg zu schicken, um Lernsituationen des Irrtums »künstlich« zu erzeugen, führt zu Motivationsproblemen und zur Störung des pädagogischen Verhältnisses.

---

[111] Wir ziehen hier den Begriff des Irrtums dem Begriff des Fehlers vor. Fehler machen können auch Maschinen, während der Begriff des Irrtums nur auf Menschen beziehbar ist.

- Der systematische, in einer Sackgasse endende Irrtum erfordert das komplexeste didaktische Handeln, da dieses in nichts weniger als in der **Bewußtmachung der Rekonstruktion und Kritik des Lösungswegs** des Lernenden für den Lernenden besteht. Dazu ist der antizipierte Lösungsweg einschließlich der Problemsicht des Lernenden durch entsprechende Fragetechniken offenzulegen. Diese Fragen beziehen sich in der Regel auf folgende handlungsstrukturierende Dimensionen:

  1. Dimension: die **subjektive Kohärenz des Lösungshandelns** des Lernenden. Es ist fragend festzustellen, ob die Lernenden ihren eigenen Prämissen (Sinngehalten) entsprechend handeln, ob sie sich also »sinnrational« verhalten.
  2. Dimension: die **Rekonstruktion der einzelnen Handlungsmotive**. Durch Fragen kann das dem Handeln der Lernenden implizite Wollen transparent und damit diskutierbar gemacht werden.
  3. Dimension: **Wahrheitsgehalt des angewandten Wissens**. Durch Fragen können die Wissenskomponenten des Lernenden auf ihre Stichhaltigkeit untersucht werden.
  4. Dimension: **Logik des irrtümlichen Lösungswegs**. Der Lehrende kann erfragen, welche Ziele (eigentlich) zu dem gegebenen Handeln und Wissen passen (im Gegensatz zu den subjektiv intendierten Zielen), wenn dieses nicht generell falsch ist.

  Diese Fragen, die wir auch als **»Moderieren aus einer Sackgasse«** bezeichnen, sind dem Lernenden nicht per se verständlich, sondern müssen, oftmals auch in Form hypothetischer Aussagen, formuliert werden (»Waren dieses die Ziele, die du dir vorgestellt hattest?«).

4.3 Zusammenfassung: Ganzheitliche Arbeitsgestaltung und der didaktische Umgang mit Wissen

Die Umsetzung des Qualifizierungsziels Ganzheitliche Arbeitsgestaltung erfordert einen differenzierten didaktischen Umgang mit Wissen. Die notwendige Integration von Technikgestaltungskompetenz, Prozeßkompetenz und Kooperationskompetenz im modernen Software-Entwicklungsprozeß kann nicht mehr durch lineares oder multifunktionales Wissen bewältigt werden. Der Prozeß entspricht weder einer linearen Abbildung von Wissen, noch einem sich verzweigenden Flußdiagramm mit feststehenden Entscheidungspunkten. Vielmehr ist nicht nur das Softwareprodukt Ausdruck von letztendlich subjektbezogener »Reality Construction«, sondern der Software-Entwicklungsprozeß selbst kann

nur als eine von Subjekten konstruierte, nicht formalisierbare Realität betrachtet werden.

Diese Sichtweise des Softwareprodukts und des Software-Entwicklungsprozesses kann die handelnden Subjekte, die IT-Fachkräfte, nicht mehr als Informationsträger fassen, sondern nur noch als epistemologische Subjekte. Die von uns vorgeschlagene Systematik wissensbezogener didaktischer Handlungsorientierungen kann als Versuch angesehen werden, **dem Lernenden als epistemologischem**[112] **Subjekt gerecht zu werden und sie nicht auf Informationsträger zu reduzieren**. In einer Systematik des wissensbezogenen didaktischen Handelns haben wir drei unterschiedliche didaktische Handlungsfelder bestimmt und ihre - für unser Anliegen - relevanten Aspekte erörtert: die Gestaltung der Lernaufgaben, die Wissensvermittlung und die Wissensmoderation.

Bei der Bestimmung des wissensbezogenen didaktischen Handelns sind wir von der Grundthese ausgegangen, daß eine Fachdidaktik, die den Anspruch hat, sich auf den Lernenden als epistemologisches Subjekt zu beziehen, nur durch die Hinzunahme letztendlich wissenschaftsphilosophischer Fragestellungen befriedigend formuliert werden kann. Die damit einhergehende Distanz zum praktischen Unterricht und damit die Weigerung, Rezepte für den Unterricht zu liefern, schmälert u.E. keineswegs die Praxisrelevanz der von uns aufgestellten didaktischen Maximen. Im Gegenteil: Konkrete Unterrichtsanweisungen, die den Ausbildungskontext nicht berücksichtigen, entpuppen sich spätestens bei der praktischen Umsetzung als fragwürdig. Da das Qualifizierungsziel »Ganzheitliche Arbeitsgestaltungskompetenz« auch den vorhandenen Ausbildungskontext in Frage stellt, wäre nur ein kontextunabhängiger Entwurf eines Unterrichtsmodells möglich gewesen, der keinerlei konkrete Ausbildungsbasis gehabt hätte. Die von uns aufgestellten Aspekte der Gestaltung von Lernaufgaben, der Wissensvermittlung und -moderation können dagegen als **Anhaltspunkte der Transformation des aktuellen Unterrichts und seines Kontextes** praktisch genutzt werden.

Neben dem Umgang mit Wissen muß sich das didaktische Handeln auch auf das Training sozialer Kompetenzen beziehen, um dem Qualifizierungsziel Ganzheitliche Arbeitsgestaltung gerecht zu werden.

---

[112] Wir verwenden hier den Fachbegriff der Epistemologie wegen seiner spezifischen philosophischen und psychologischen Implikationen. Die Aussage verliert aber nicht an Bedeutung, wenn wir von dem Lernenden als aktiv erkennendem Subjekt sprechen: einem Subjekt, das sich die Welt aneignet indem es sie entwirft, und keinem Subjekt, das die Wirklichkeit als »vorgekaute« Lehrmeinung aufsaugt.

5 Didaktisches Handeln und das Training sozialer Kompetenzen

Während das wissensbezogene didaktische Handeln sich auf den Lernenden als epistemologisches Subjekt bezieht, betrachtet das auf soziale Kompetenzen ausgerichtete didaktische Handeln die Lernenden als kommunikativ interagierende Subjekte. Als Training wird systematisches Üben bezeichnet. Wenn wir uns im folgenden den Möglichkeiten des Trainings sozialer Kompetenzen im Kontext der Ausbildung nähern, müssen wir uns folgendes vor Augen führen:

- Soziale Kompetenzen können nur in der **Gruppe** trainiert werden.
- Wichtigstes didaktisches Gestaltungsmedium ist die Entwicklung von **Unterrichtsformen**, die sozialen Kompetenzen förderlich sind.
- Aktives didaktisches Handeln besteht darin, dem Lernenden ein **Feedback** bezüglich seines kommunikativen Handelns durch Selbstkonfrontation, Gruppenkonfrontation und Handlungsanalyse (insbesondere Sprechaktanalyse) zu geben.

Als prinzipielle Einschränkung des didaktisch-pädagogischen Handelns muß gleichzeitig folgendes berücksichtigt werden:

- Das Training von sozialen Kompetenzen setzt immer schon das Vorhandensein von weitgehend entwickelten sozialen Kompetenzen voraus. **Der Lehrende kann nicht die Position eines Therapeuten einnehmen**, der soziale Kompetenzdefizite systematisch therapiert.
- **Das Training sozialer Kompetenzen muß in einer sachlichen Beziehung zum Software-Entwicklungsprozeß stehen.** Das losgelöste Training von sozialen Kompetenzen entspricht nicht der berufsbezogenen Lernmotivation der Lernenden.

Das heißt für uns: Das Ziel des Trainings sozialer Kompetenzen kann als Orientierungsziel des didaktischen Handelns nicht therapeutisch formuliert werden. Vielmehr geht es darum, daß in der Ausbildung Lernsituationen geschaffen werden, die der Entfaltung sozialer Kompetenzen dienen, und daß die Lernenden Regeln verstehen und verinnerlichen, die ihnen eine Selbstkontrolle ihres kommunikativen Handelns ermöglichen.

Im folgenden werden wir uns daher auf zwei Formen des nichttherapeutischen didaktischen Handelns beschränken, die sich zur Förderung sozialer Kompetenzen in der Ausbildung einsetzen lassen:

- **Kommunikationsregeln** in der Gruppenarbeit einführen, erarbeiten und einüben.

- Sozialkompetenzen fördernde **Unterrichtsformen** gestalten.

5.1 Kommunikationsregeln in der Gruppenarbeit

Die Förderung sozialer Kompetenzen durch die Einführung, Erarbeitung und Einübung von Kommunikationsregeln in der Gruppe, beziehungsweise in der Gruppenarbeit, ist wahrscheinlich die verbreitetste Form didaktischen Handelns, das sich auf soziale Kompetenzen bezieht. Bevor wir allerdings diese Kommunikationsregeln im einzelnen darstellen werden, erscheint es uns notwendig, auf die Tragweite des didaktischen Konzepts von Kommunikationsregeln einzugehen.

Als soziale Kompetenzen haben wir die Frustrations- und Ambiguitätstoleranz, die Rollendistanz, Sprachkompetenz und Empathie bezeichnet. Diese Bestimmung der sozialen Kompetenzen zeichnet sich nicht nur durch ihren anthropologischen Bezug zum Menschen als Gattungswesen aus, sondern auch durch eine Prozeßhaftigkeit, die zwischen den Polen Handlungsoffenheit (-freiheit) und sozial determinierter Situationsangemessenheit des Handelns oszilliert. Diese hochkomplexe Prozeßhaftigkeit personengebundener sozialer Kompetenzen auf formale Kommunikationsregeln didaktisch zu verkürzen, ist u.E. dann ein zweifelhaftes Unternehmen, wenn der Formcharakter der Kommunikationsregeln im Vordergrund steht. Das heißt für uns: Die nachfolgend vorgestellten Regeln sind didaktisch nicht als Formalismen einzuführen, sondern als kritisierbare Regeln, die ihre Legitimation erst in den einzelnen Kommunikationssituationen durch die Kommunikationsteilnehmer erfahren. Ihre Formgestalt haben sie nur insofern, als sie Aspekte eines normativen, auf Verständigung orientierten Kommunikationsprozesses abbilden. Ohne diese Grundorientierung auf eine tendenziell herrschaftsfreie Verständigung verlieren die Kommunikationsregeln den Prozeßbezug, der sie letztendlich generiert und reproduziert. Auch Kommunikationsregeln sind keine didaktischen Methoden, die jenseits des pädagogischen Verhältnisses soziale Kompetenzen fördern können. Das konkrete personale pädagogische Verhältnis bildet gewissermaßen die »materielle« prozeßhafte Basis für die Etablierung expliziter Kommunikationsregeln.[113] Im folgenden

[113] Insbesondere hier werden die Probleme des pädagogischen Verhältnisses deutlich: Die funktionale Organisation der Ausbildung, ihre Selektions- und Allokationsfunktion, bietet keine Basis für an Verständigung orientierte Kommunikationsregeln, sondern Anlaß, Kommunikationssituationen unter taktischen Gesichtspunkten zu betrachten und zu bewältigen. Auf der anderen Seite ist ein Lehrer-Schüler-Verhältnis, das die reale personale Asymmetrie zwischen Lehrendem und Lernenden leugnet, ebensowenig geeignet, diese Kommunikationsregeln sich wirklich zu eigen zu machen. Nur der Lehrende, der die Organisationsfunktion und die personale Beziehung

werden wir einige Vorschläge zur didaktischen Verwendung von Kommunikationsregeln vorstellen und näher erläutern.

5.1.1 Kommunikationsregeln als didaktische Methode der Negation

Ausgangspunkt der didaktischen Methode der Negation ist hier die Tatsache, daß jegliche Kommunikation, so unbefriedigend sie auch ist, durch Regeln bestimmt ist. Die Aufdeckung und Bearbeitung vorhandener Kommunikationsregeln ist die Voraussetzung zur Etablierung neuerer, den Zielen der Kommunikation angemessenerer Regeln. Nach Fleischer ist es sinnvoll, zwischen vier Grundtypen von Kommunikationsregeln zu unterscheiden [vgl. Fleischer 1990, S. 90]. Diese Unterscheidungen kommen dadurch zustande, daß nach der Funktionalität, beziehungsweise Dysfunktionalität von Regeln bezüglich der angestrebten Ziele gefragt wird, und daß zwischen sprachlich kodierten (expliziten) und sprachlich nicht-kodierten (impliziten) Regeln differenziert wird. Es ergibt sich dann folgende Matrix:

	kodiert	nicht-kodiert
funktional	1.	2.
dysfunktional	3.	4.

Zur Illustration wollen wir jeweils ein Beispiel nennen, das die Lehrer-Schüler-Kommunikation betrifft.

zu 1.:

Sprich im Unterricht nicht ohne Erlaubnis! Diese Regel ist im »normalen« Unterricht durchaus sinnvoll. Wenn sie allerdings totalisiert wird, kann sie Kommunikation dysfunktional stören, z.B. in der Gruppenarbeit. Ebenfalls kann es sinnvoll sein, daß der Schüler den Lehrer auch unerlaubt unterbricht, wenn dieser beispielsweise im Unterrichtsvortrag fortfährt, ohne auf die Verstehensprobleme der Schüler einzugehen.

Funktional kodierte Kommunikationsregeln müssen bezüglich ihres Geltungsbereichs reflektiert und kritisiert werden.

---

im pädagogischen Verhältnis aufhebt, kann Kommunikationsregeln sinnvoll einsetzen (Gerechtigkeit, personale Achtung und fachliche Kompetenz gelten als die wesentlichen Zuschreibungen an die Person des Lehrenden durch die Lernenden, die über die Qualität des pädagogischen Verhältnisses entscheiden).

zu 2.:

Greife nie jemanden in seiner ganzen Person an! Die personale Achtung ist für die Aufrechterhaltung von Kommunikation unerläßlich. Die Beachtung dieser Regel wird allerdings dann dysfunktional, wenn sie dazu führt, daß einzelne Verhaltensweisen von Personen nicht kritisiert werden dürfen.

Funktional nicht-kodierte Kommunikationsregeln müssen reflektiert und damit sprachlich kodiert werden, um ihren Geltungsbereich zu bestimmen.

zu 3.:

Melde dich nur, wenn du sicher bist, daß du etwas Richtiges weißt! (Stelle nie dumme Fragen!) Diese Regel limitiert Kommunikation derart, daß sie kaum mehr zur dialogischen Unterstützung des Lernprozesses beitragen kann. Die Überwindung dieser Regel durch die Regel »Es gibt keine dummen Fragen und Antworten!« kann didaktisch sinnvoll sein, obwohl sie inhaltlich sicherlich zweifelhaft ist.

Dysfunktional kodierte Regeln sind kritisch zu betrachten und durch verbindliche Gegenregeln zu ersetzen.

zu 4.:

Kritisiere niemals einen Stärkeren, denn wer stärker ist, setzt sich durch! Diese oftmals auch die Schüler-Schüler-Kommunikation bestimmende Regel ist auch in ihren verfeinerten Formen zu kodieren und damit der Kritik zugänglich zu machen. Wobei u.E. insbesondere darauf zu achten ist, daß die Überwindung dieser Regel nicht durch die Ausblendung realer Machtverhältnisse erfolgt (Gerade in dem Software-Entwicklungsprozeß ist in der Regel die Gestaltungsmacht zwischen Entwicklern und Benutzern ungleich verteilt, obwohl gleichberechtigt gestaltet werden sollte. Die regulative Regelung dieser Machtungleichheit mit Hilfe von Kommunikationsregeln ist daher essentiell.).

Dysfunktional nicht-kodierte Regeln sind bewußt zu machen, auf ihren realen Hintergrund zu durchleuchten und durch verbindliche Gegenregeln zu ersetzen.

Da das Aufspüren von funktionalen und dysfunktionalen, kodierten und nichtkodierten Regeln der Kommunikation für die im Kommunikationssystem Eingebundenen äußert schwierig ist, stellt sich die Frage, von welchem archimedischen Punkt diese zu bestimmen sind. Um die Antwort vorwegzunehmen: Einen solchen Punkt gibt es nicht. Es gibt nur theoretische Modelle und Idealvorstellungen der Kommunikation, die Anhaltspunkte für die Analyse bestehender Regeln und ihrer konstruktiven Überwindung bieten. Im folgenden wollen wir als ein

solches Modell die pragmatischen Axiome der Kommunikation nach Watzlawick vorstellen. Im Anschluß daran werden wir formale Kommunikationsregeln, wie sie sich in der Diskussion um die Themenzentrierte Interaktion gebildet haben, kurz skizzieren.

5.1.2 Pragmatische Axiome der Kommunikation und die vier Aspekte einer Nachricht

Watzlawick u.a. haben in einer vielgelesenen Studie Ende der sechziger Jahre pragmatische Axiome der Kommunikation aufgestellt, die zum Ausgangspunkt vieler praktisch orientierter Kommunikationsregeln geworden sind:

Erstes Axiom: "*Man kann nicht* nicht *kommunizieren*" [Watzlawick u.a. 1972, S. 53]

Dieses Axiom besagt, daß jegliches menschliche Verhalten in einer Kommunikationssituation Mitteilungscharakter hat. Da es kein Nicht-Verhalten gibt, gibt es auch keine Nicht-Kommunikation. Kommunikation kann demnach keineswegs auf absichtsvolle Kommunikation beschränkt werden. Wenn aber jeglichem Verhalten ein kommunikativer Charakter zukommt, so ist es auch durch die weiteren Axiome näher bestimmbar.

Zweites Axiom: "*Jede Kommunikation hat einen Inhalts- und einen Beziehungsaspekt, derart, daß letzterer den ersteren bestimmt und daher eine Metakommunikation ist* " [Watzlawick u.a. 1972, S. 56][114]

Schulz von Thun hat diesen doppelten Nachrichtenaspekt nochmals ausdifferenziert [vgl. Fleischer 1990, S. 157ff.]. Neben dem Sachinhalt und der Beziehung hat Kommunikation auch einen **Selbstoffenbarungs- und Appellcharakter**. Der »nachrichtliche« Aspekt einer Kommunikation läßt sich demnach durch vier Fragestellungen untersuchen:

- Sachinhalt: Worüber informiere ich?
- Selbstoffenbarungsaspekt: Was gebe ich von mir selbst kund?
- Beziehungsaspekt: Was halte ich vom anderen, wie stehen wir zueinander?

---

[114] Watzlawick u.a. haben dieses Axiom mit folgender Anmerkung versehen: "In diesem Definitionsversuch nehmen wir etwas arbiträr an, daß der Beziehungsaspekt den Inhalt determiniert oder subsumiert, obwohl es logisch ebenso richtig wäre, zu sagen, daß eine Klasse (Menge) von ihren Elementen - und daher die Beziehung vom Inhaltsaspekt - bestimmt wird. Da unser Hauptinteresse aber die metakommunikativen Aspekte der Pragmatik und weniger die Eigenschaften des Informationsaustausches sind, ziehen wir die oben genannnte Formulierung vor" [Watzlawick u.a. 1972, S. 56].

- Appellaspekt: Wozu möchte ich den anderen veranlassen?

Dadurch, daß eine Nachricht oftmals alle vier Aspekte beinhaltet, die Kommunikationsteilnehmer aber selektiv nur bestimmte Aspekte wahrnehmen, aber auch mitteilen wollen, kommt es zu verzerrter Kommunikation.

Drittes Axiom: "*Die Natur einer Beziehung ist durch die Interpunktion der Kommunikationsabläufe seitens der Partner bedingt*" [Watzlawick u.a. 1972, S. 61]

Dieses Axiom besagt, daß Kommunikation von außen betrachtet zwar wie ein ununterbrochener Fluß erscheint, die einzelnen Kommunikationsteilnehmer ihre Kommunikationsbeiträge immer aber auf ganz bestimmte Kommunikationsereignisse beziehen müssen. Jeder Kommunikationspartner sieht den Beginn eines Kommunikationsprozesses in einem bestimmten Anfangspunkt, von dem aus sich die Kommunikationsstruktur für ihn entwickelt. Wenn die Interpunktionen der Kommunikationsabläufe stark differieren, wird Kommunikation als Verständigung unmöglich. Dies erleben wir oft bei Konflikten, wenn beide Konfliktpartner ihr eigenes Kommunikationsverhalten nur als Reaktion auf das Verhalten ihres Gegenüber ansehen und gar nicht bemerken, daß ihre Kommunikationsbeiträge die Interpunktionsstruktur des anderen bestimmen. Die distanzierte Betrachtung der eigenen Interpunktionsstruktur ist oftmals Grundlage dafür, den Konflikt überhaupt angemessen zu erfassen und damit gegebenenfalls lösbar zu machen.[115]

Viertes Axiom: "*Menschliche Kommunikation bedient sich digitaler und analoger Modalitäten*" [Watzlawick u.a. 1972, S. 68]

Dieses Axiom benennt die Differenz zwischen der Kommunikation mit abstrakten, formalisierten Zeichen, deren Bedeutung allein durch eine soziale Vereinbarung zustandekommt, und Zeichen, Gebärden, Betonungen usw., deren Bedeutung durch Analogien erschließbar ist. Wichtig ist für unseren Zusammenhang, daß sich analoge Modalitäten insbesondere auf den Beziehungsaspekt, den Selbstoffenbarungsaspekt und den Appellaspekt von Kommunikation konzentrieren. Didaktisch-methodisch ist die Diskrepanz entscheidend, die sich oftmals zwischen verbaler (digitaler) und nonverbaler (analoger) Kommunikation, z.B. bei der Annahme einer Lernaufgabe durch die Schüler, zeigt. Die Transformation analoger Mitteilungen in digitale Sprache ist allerdings die einzige Möglichkeit, sie einem intersubjektiven Reflexionsprozeß bewußt zu machen. Die Probleme,

---

[115] Wenn die unterschiedlichen Interpunktionsstrukturen einen grundsätzlichen Interessenskonflikt darstellen, kann diese Distanz zumindest zu einer Offenlegung dieser Interessen führen.

die sich aus der **Diskrepanz von analoger und digitaler Kommunikation** ergeben, sind für zukünftige IT-Fachkräfte keineswegs nur von inhaltlicher Bedeutung, insofern sie Kommunikationsmedien gestalten, die prinzipiell nur digitalisierte Kommunikation transportieren[116]. Vielmehr werden wir weiter unten deutlich machen, daß das Verhältnis zwischen analoger und digitaler Kommunikation im beruflichen Handeln maßgeblich auch durch die berufliche Identität bestimmt wird. Das Aufdecken und Reflektieren dieser Diskrepanzen wird uns bei dem didaktischen Handeln, das sich die Entwicklung einer angemessenen beruflichen Identität zum Ziel setzt, näher beschäftigen.

Fünftes Axiom: "*Zwischenmenschliche Kommunikationsabläufe sind entweder symmetrisch oder komplementär, je nachdem, ob die Beziehung zwischen den Partnern auf Gleichheit oder Unterschiedlichkeit beruht*"
[Watzlawick u.a. 1972, S. 71]

Komplementäre Interaktion basiert "auf sich gegenseitig ergänzenden Unterschiedlichkeiten" [Watzlawick u.a. 1972, S. 69]. Die Lehrer-Schüler-Kommunikation ist ebenso als prinzipiell komplementär zu bezeichnen wie die Entwickler-Benutzer-Kommunikation. Da Komplementarität nach Watzlawick "die ineinander verzahnte Natur der Beziehung" [ebd., S. 70] widerspiegelt, können wir allerdings weder bei der Lehrer-Schüler-Kommunikation noch bei der Entwickler-Benutzer-Kommunikation von einem rein komplementären Kommunikationsablauf ausgehen. Während die Lehrer-Schüler-Kommunikation durch das Erziehungsideal der Selbständigkeit und durch den gesellschaftlichen Bildungsauftrag immanent auf die Überwindung der Komplementarität ausgerichtet ist, fehlen der Entwickler-Benutzer-Kommunikation die komplementären gesellschaftlichen Verhaltenserwartungen. Die Lehrer-Schüler-Kommunikation muß daher auch immer Kommunikationssituationen kennen, in denen ein symmetrischer Kommunikationsablauf erwünscht ist, weil nur so das Lernziel Selbständigkeit erreichbar ist. Andersherum verhält es sich bei der Entwickler-Benutzer-Kommunikation. Der Benutzer entwickelt seine komplementäre Rolle erst im Projekt, das heißt, die Herstellung der Komplementarität ist das eigentliche Problem. Diese Komplementarität beruht gleichzeitig immer auf einem symmetrischen Verhältnis, der Kommunikation zwischen Experten.

---

[116] Im Bereich der Multimedia-Technik werden allerdings auch Bilder und akustische Signale digital übermittelt, die im Sinne Watzlawicks einen analogen Modus aufweisen. Die Chancen und (Manipulations-)gefahren, die diese technischen Möglichkeiten bieten, verlangen von den IT-Fachkräften ebenfalls ein differenziertes, reflektiertes Verständnis menschlicher Kommunikation.

5.1.3 Die pragmatischen Axiome der Kommunikation und die didaktische Förderung sozialer Kompetenzen durch Reflexion und Kommunikationsregeln[117]

Die pragmatischen Axiome als Lernstoff zu vermitteln, mag zwar begrenzten didaktischen Sinn machen, relevante Bedeutung gewinnen sie allerdings nur, wenn sie mit **realen Kommunikationserfahrungen** verbunden werden. Da die Axiome den Anspruch haben, jegliche Kommunikation axiomatisch zu erfassen, scheint es nichts Leichteres zu geben, als diesen Erfahrungsbezug herzustellen. Kommunikation findet ständig statt. Beispiele für die Axiome gibt es zuhauf. Aber gerade dadurch, daß wir ständig kommunizieren, sehen wir nicht, wie wir kommunizieren. Erst wenn Kommunikation auch für die Kommunizierenden problematisch geworden ist, besteht die didaktische Möglichkeit, Kommunikation zum Gegenstand der Reflexion zu machen. Solange die Kommunikation in der **ritualisierten** Form des Frontalunterrichts verbleibt, gibt es wenig Ansatzpunkte, sie zu reflektieren.

Das heißt für uns: Das Aufbrechen ritualisierter schulischer Kommunikationsformen ist die didaktische Voraussetzung für die Erfahrung der Kommunikation als problematischem und damit problematisierbarem Gegenstand. Positiv ausgedrückt: Es geht um die Schaffung **»evidenter Lernsituationen«**, in denen die pragmatischen Axiome situativ von offensichtlicher Bedeutung sind und **gleichzeitig** handlungsleitende, sprich kompetenzfördernde Bedeutung für die Lernenden erlangen. Auf die didaktischen Gestaltungsprämissen, die evidente Lernsituationen wahrscheinlicher machen, werden wir weiter unten unter dem Stichwort »Lernsetting« näher eingehen. Daß sich diese Lernsituationen kaum im ritualisierten Frontalunterricht ergeben, sondern eher in **didaktischen Formen des Gruppenlernens** (Plenumsdiskussion, Rollenspiele, Projektarbeit), setzen wir als selbstverständlich voraus und wollen es deshalb hier nicht näher begründen.

---

[117] Alle im folgenden mit Anführungszeichen versehenen Kommunikationsregeln sind im wesentlichen ursprünglich von Ruth Cohn als Regeln der Themenzentrierten Kommunikation aufgestellt worden. Die Selbstregulation der Gruppe als Verhältnis von Ich-Wir-Thema und die sozialen Lernfähigkeiten der einzelnen Gruppenmitglieder sind Ziele dieser Regeln (Eine ausführliche Besprechung findet sich bei Fleischer 1990, S. 92ff.). Fleischer weist ebenfalls darauf hin, daß diese Regeln zwar zur Herstellung einer erstrebenswerten Interaktionskultur von größter Bedeutung sind, aber keineswegs vollständig und für jede Interaktion angemessen sind [Fleischer 1990, S. 93f]. Wir haben uns hier deshalb für eine eklektizistische Verwendung dieser Kommunikationsregeln entschieden, die allerdings dadurch eine neue Qualität erhalten, daß wir sie beispielhaft mit den pragmatischen Axiomen Watzlawicks verknüpfen.

Die pragmatischen Axiome haben in einer »evidenten Lernsituation« **zwei Formseiten**. Die eine Seite bezieht sich auf **Kommunikationsereignisse**. Die andere Seite bezieht sich auf die **sozialen Kompetenzen** der handelnden Subjekte. Zum Beispiel: In einem Rollenspiel reden zwei Konfliktpartner immer aneinander vorbei. Keiner der Beteiligten versteht, warum der andere ihn nicht versteht. Der Konflikt scheint auf der inhaltlichen Ebene nicht lösbar zu sein. Der Lehrer bietet sich nun als **Kommunikationsmoderator** an und rekonstruiert gemeinsam mit den Konfliktpartnern die jeweiligen Kommunikationsereignisse, die als Interpunktionen den jeweiligen subjektiven Kommunikationsablauf gebildet haben. Die gemeinsame Rekonstruktion unterstützt die Herausbildung eines dezentrierten Vorstellungssystems, das wiederum den kognitiven Teil der sozialen Kompetenz Empathie ausmacht. Das Rekonstruktionsergebnis verweist wiederum oftmals auf Differenzen zwischen dem Kommunikationsziel (Kommunikationsinteressen) und dem tatsächlichen Kommunikationsablauf. Die Diskussion darüber, wie der einzelne seine Interessen verständlich hätte kommunizieren können, fördert die Sprachkompetenz des Lernenden.

Gleichzeitig kann auch herausgearbeitet werden, daß einzelne Kommunikationsprobleme durch spezifische rollengebundene **Kommunikationsregeln** verstärkt werden. Zum Beispiel konnte ein Kommunikationsteilnehmer sich kaum mehr auf das Thema konzentrieren, weil er dem anderen noch etwas anderes mitteilen wollte, was gar nicht zur rollenspezifischen Kommunikation gehörte. Die Regel »Kommuniziere in einer sachlichen Kommunikation nur etwas, das sich auf die Sache bezieht« kann nun kritisch hinterfragt werden. Die Gegenregel »Störungen haben Vorrang« kann auf ihre praktische Bedeutung untersucht werden. Die Auseinandersetzung mit diesen Regeln fördert wiederum die soziale Kompetenz der Rollendistanz, weil Rollendistanz nun nicht mehr als Ausscheren aus der Rollenkommunikation angesehen wird, sondern als Voraussetzung für die Aufrechterhaltung von rollengeleiteter Kommunikation gilt.

Die pragmatischen Axiome können als heuristisches didaktisches Mittel eingesetzt werden, das in »evidenten Lernsituationen« zwischen den Kommunikationsereignissen und den sozialen Kompetenzen vermittelt:

Kommunikationsereignisse ⇔ **pragmatische Axiome** ⇔ **soziale Kompetenzen**

Wir behaupten sogar, daß die pragmatischen Axiome der Kommunikation den **didaktischen Handlungsbereich »Soziale Kompetenzen«** umfassend fundieren. Selbst die Konzeption kooperativer Unterrichtsformen muß bei der Antizipation »evidente Lernsituationen« sozialer Kompetenzen auf die pragmatischen Axiome der Kommunikation als Reflexionsbasis zurückgreifen.

Da die **Verwendung der pragmatischen Axiome** sowohl als Reflexionsbasis des Lehrenden als auch als Reflexionswissen in der Kommunikation zwischen den Lernenden **nur situativ ausgestaltet** werden kann, ist es uns hier nicht möglich, didaktisch-technische Formen ihrer Verwendung aufzulisten. Diese bleiben der sozialen Kreativität des Lehrenden vorbehalten. Wir können hier also nur beispielhaft mögliche Zusammenhänge skizzieren:

Das Axiom der Komplementarität und soziale Kompetenzen

Das Axiom der Komplementarität versus Symmetrie der Kommunikation ist engstens mit der sozialen Kompetenz der Rollendistanz verbunden. Wir haben schon weiter oben auf die Bedeutung dieses Axioms in der Kommunikation zwischen Entwicklern und Benutzern hingewiesen. Wenn sich die prinzipiell komplementäre Rollenkommunikation zwischen Entwicklern und Benutzern nur in Kommunikationsereignissen eines Frage-Antwort-Schemas widerspiegelt, liegt u.E. eine verzerrte Kommunikation vor. Der jeweilige Kommunikationspartner erscheint so nur als Träger von Informationen, deren Sinngehalt für die Projektaufgabe diesem selbst verschlossen bleibt (der Benutzer hat demnach die Informationen über die Arbeit, aber er kann den Sinn dieser Informationen für die Gestaltung eines neuen Arbeitssystems nicht einsehen).

Unser Verständnis von Systementwicklung geht hingegen von einer **Balance von komplementärer und symmetrischer Kommunikation** aus: Komplementäre Expertenkommunikation ist symmetrisch, insofern sie auf die personale Achtung des anderen als Experten und auf der Einsicht beruht, daß kein Experte bezüglich des Projektziels alleiniger Experte ist. Diese Balance von komplementärer und symmetrischer Kommunikation ist in einer durch komplementäre Kommunikationsereignisse gekennzeichneten Situation nur durch Rollendistanz zu erhalten. Das Aufbrechen komplementärer Kommunikation durch **Kommunikationsregeln symmetrischer Kommunikation** fördert die soziale Kompetenz der Rollendistanz. Solche Regeln können zum Beispiel sein: »Äußere deine Meinung, statt Fragen zu stellen - verstecke Meinungen nicht hinter Fragen!« oder »Gebe Rückmeldungen und nehme Rückmeldungen an!« und »Sprich per 'Ich', nicht per 'MAN', 'Wir' oder 'ES'!«. Die Erarbeitung solcher Kommunikationsregeln in evidenten Lernsituationen ist, ebenso wie die gemeinsame Reflexion von symmetrischen und komplementären Kommunikationsabläufen, ein didaktisches Mittel, um Rollendistanz indirekt zu fördern.

Das Axiom der vier Nachrichtenaspekte und soziale Kompetenzen

Das pragmatische Axiom, daß jede Kommunikationseinheit einen vierfachen Nachrichtenaspekt hat, ist vielfältig mit sozialen Kompetenzen verknüpft. Es verweist prinzipiell auf die Sprachkompetenz, die Rollendistanz, die Empathie und die Ambiguitätstoleranz. Insbesondere ist in »evidenten Lernsituationen« deutlich zu machen, daß auch eine scheinbar rein zweckrationale Kommunikation einen Selbstoffenbarungsaspekt, einen Beziehungsaspekt und einen Appellaspekt hat. Wenn ich über etwas informiere, gebe ich auch etwas von mir kund, zeige ich auch, was ich vom anderen halte und wie wir zueinander stehen und möchte den anderen auch zu irgendetwas veranlassen. Allerdings werde ich mich bei einem Kommunikationsakt auf einen Nachrichtenaspekt konzentrieren und nur selten alle Aspekte gleichwertig verfolgen. Andersherum rezipiere ich auch Kommunikationsereignisse mehr oder weniger bewußt als »vierohriger Empfänger« [Fleischer 1990, S. 162], wobei ich mich meist auf einen Aspekt konzentriere.

Während der sachliche Inhalt einer Information sprachanalytisch relativ exakt bestimmt werden kann (als deiktische Handlung, als Definition, als logische, normative oder empirische Aussage)[118], so ist weder der »**gemeinte« Nachrichtenaspekt** eindeutig bestimmbar, noch sind der Selbstoffenbarungs-, Beziehungs- und der Appellaspekt aus dem Kommunikationsereignis allein rekonstruierbar. Schulz von Thun[119] gibt als Beispiel dafür den einfachen Satz »Du, da vorne ist grün!« eines Beifahrers als Kommunikationsereignis an. Neben der banalen Sachinformation kann der Satz die verschiedensten Bedeutungen haben: Neben der »objektiven« Selbstoffenbarung des Beifahrers, daß er deutschsprachig und farbsichtig ist, kann er mit dieser Aussage auch kundtun, daß er es eilig hat. Der einfache Satz kann auch einen Beziehungsaspekt meinen: Im Beispiel Schulz von Thuns antwortet die fahrende Frau: »Fährst du oder fahre ich!?« Das heißt, sie versteht seinen Satz als Beziehungsdefinition, die lautet: »Ich bin der, der dir nicht zutraut, ohne Hilfe zu fahren!« Darüber hinaus kann der

---

[118] Mit der Behauptung, daß sprachanalytisch eine relativ exakte Bestimmung des sachlichen Inhalts einer Information möglich ist, wollen wir keineswegs suggerieren, daß diese Sprachanalyse einfach ist. Neben der Bestimmung des Geltungsbereichs von Aussagen steht nicht weniger als die Bestimmung der Gültigkeit von Aussagen. Die deiktische Handlung "Das sehen Sie hier!" kann äußerst fragwürdig sein, wenn sie sich nicht auf die Realität, sondern auf Artefakte oder Realitätsmodelle (z.B. ein Entity-Relationship- Modell) bezieht. Desweiteren entpuppen sich auch Definitionen, die scheinbar exakt und klar sind, bei näherer Betrachtung als zirkulär in dem Sinne, daß das zu Definierende (definiendum) im Definierenden (definiens) enthalten ist.

[119] Vgl. ausführlich zu diesem Beispiel: [Fleischer 1990, S. 159ff.].

Satz »Du, vorne ist grün!« den Apellcharakter »Gib Gas!« haben. Die Schwierigkeit der Kommunikation besteht nun darin, sowohl **auf einer Ebene der Nachrichtenaspekte zu kommunizieren**, als auch **das »Gemeinte« zu verstehen**.

Die Besonderheit der Benutzer-Entwickler-Kommunikation liegt in ihrer rollenkommunikativen Verfaßtheit, die nur auf **sachlich-inhaltliche Kommunikationsereignisse** ausgerichtet ist, obwohl der Software-Entwicklungsprozeß aber als **sozialer Prozeß** alle Nachrichtenaspekte notwendig macht. Das heißt, Entscheidungen im Entwicklungsprozeß sind nicht allein sachlich-inhaltlicher Natur, sondern sie kommen erst durch soziale Konflikte zustande. Ein starres Festhalten an dem sachlich-inhaltlichen Nachrichtenaspekt in der Rollenkommunikation führt zu Verzerrungen der Kommunikation, die diese selbst gefährden. Sachfragen werden zur Klärung von Machtfragen, zu Selbstdarstellungen und zum Handlungsappell verwendet, ohne daß dieser Gebrauch kontextuell von wirklichen Sachfragen abgrenzbar wäre. Die Überwindung dieser Kommunikationsverzerrungen erfordert soziale Kompetenzen, die durch die didaktische Umsetzung des pragmatischen Axioms der vier Nachrichtenaspekte gefördert werden.

Die Beachtung des Nachrichtenaspekts von Kommunikationsereignissen kann didaktisch durch **gemeinsames Reflektieren** unterstützt werden. Didaktischer Ausgangspunkt dieser Reflexion kann die **Rekonstruktion einer konflikthaften Kommunikation** oder ein **spontanes Feedback** (Benennung von vermuteten Nachrichtenaspekten) des Lehrers als Kommunikationsmoderator bilden. Auch hier gibt es **Kommunikationsregeln**, die der Klärung des Nachrichtenaspekts von Kommunikationsereignissen dienen, das heißt, die Mißverständnisse, die von unterschiedlichen Aspektbezügen der Kommunikationspartner herrühren, offenlegen, beziehungsweise verzerrte Konfliktlinien deutlich werden lassen. Die Regeln lauten: **»Gib Feedback, wenn du das Bedürfnis hast!**« und **»Wenn du Feedback erhältst, höre ruhig zu!«**. Beide Regeln gehen davon aus, daß durch das bewußte Einsetzen von Metakommunikationen, die die Aspektebene des Empfangens von Nachrichten explizit benennen, Verstehen wahrscheinlicher gemacht wird.

Diese Kommunikationsregeln durchbrechen den »normalen« Kommunikationsfluß systematisch. Sie können auch als systematisierte Form der Rollendistanz bezeichnet werden. Gleichzeitig fördern sie offenkundig die Empathie und die Sprachkompetenz. Wir behaupten, daß die regelgestützte aspektbezogene Metakommunikation darüber hinaus auch zur Förderung der Ambiguitätstoleranz beiträgt, weil das technische Kommunikationsparadigma, das von einer in der Sprache gegebenen Einheit von Mitteilung-Information-Verstehen ausgeht,

nachhaltig zerstört wird, aber gleichzeitig (auch emotional) positive Erfahrungen »mühsamen« Verstehens ermöglicht werden.

Das Axiom der Interpunktion von Kommunikationsabläufen und soziale Kompetenzen

Das Dilemma, daß die als Interpunktionen wirkenden Kommunikationsereignisse, auf die ich meine Kommunikationen jeweils beziehe, nicht mit der meines Kommunikationspartners identisch sind, ist unlösbar. Diese Unschärfe in der Kommunikation muß allerdings keineswegs problematisch sein, sondern schafft gerade Freiheitsgrade, die der Kommunikation oft ganz unverhoffte Wendungen geben. Wenn ich einen Text gliedere, indem ich Überschriften einfüge, einzelne Begriffe durch Fettdruck hervorhebe, tue ich im Prinzip nichts anderes, als Kommunikationsereignisse als Interpunktionen zu setzen. Der Leser, der den Text liest, bemerkt durchaus diese Interpunktionen. Er wird aber, bedingt durch sein persönliches Interesse und seine persönlichen Vorerfahrungen, andere Interpunktionen als die seinigen wählen, wenn er z.B. eine Kritik schreibt. Im Extremfall kann das dazu führen, daß ich meinen Text in der Kritik überhaupt nicht mehr wiederfinde. Ähnliches haben wir bei der Einführung des Axioms der Interpunktion anhand von Konflikten dargestellt. Tatsächlich wird auch direkte personale Kommunikation keineswegs dadurch interpunktiert, daß wir unsere aktuelle Kommunikation immer auf das jeweils gerade vorangegangene Kommunikationsereignis unseres Gegenübers beziehen.

Die Entwickler-Benutzer-Kommunikation ist nun in besonderem Maße durch Probleme der Interpunktion geprägt, da sie nicht auf einen gemeinsamen Diskurshintergrund zurückgreifen kann. Jeder hat »seinen eigenen Text« und hat Angst, ihn dadurch zu verlieren, daß er seine Kommunikationseinheiten immer auf das vorangegangene unkontrollierte Kommunikationsereignis des anderen bezieht. Der Entwickler hat gern einen Fragenkatalog im Hinterkopf, nach dem er seine Kommunikation strukturiert. Als typische Äußerung eines Entwicklers kann folgende gelten: »Wenn ich auf alles eingehen würde, was der Anwender gerade sagt, dann würde die Kommunikation im Nirgendwo enden. Ich muß genau wissen, was ich wissen will. Das kann ich dann im Gespräch abhaken.«

Die Relevanz des Axioms der Interpunktion von Kommunikationsabläufen für das didaktische Handeln liegt in erster Linie darin, daß es die Aufmerksamkeit des Lehrenden für Kommunikationsprobleme fokussiert. Diese Aufmerksamkeit ermöglicht wiederum die Rekonstruktion von Kommunikationsabläufen. Kommunikationsprobleme können so erfahrungsnah zum Gegenstand metakommunikativer Auseinandersetzung werden und somit die Fähigkeit des Schülers zur Selbstreflexion seines kommunikativen Handelns erhöhen. Im folgenden haben

wir einige typische Kommunikationsprobleme aufgelistet, die auf Interpunktionsproblemen beruhen:

- **Kommunikationskonflikte, denen ein ungeklärtes Ursache-Wirkungs-Verhältnis zugrunde liegt:** Jedes Kommunikationsereignis des Kommunikationspartners wird als Ursache des Konflikts angesehen, während die eigenen Kommunikationsakte nur als Wirkung empfunden werden. "In dieser Beziehungsform ist kein Verhalten Ursache des anderen; jedes Verhalten ist vielmehr sowohl Ursache als auch Wirkung" [Watzlawick u.a. 1972, S. 93]. Ein extremer Fall eines solchen Kommunikationskonfliktes wird mit dem Begriff der **selbsterfüllenden Prophezeiung** beschrieben. Ein Entwickler, der felsenfest davon überzeugt ist, daß die Benutzer von ihrer konkreten Arbeit nicht abstrahieren können, wird sich im Gespräch mit den Benutzern nur bestätigt fühlen, da er kommunikativ nur Abstraktes einfordert, aber nicht den Prozeß der Abstraktion gemeinsam mit dem Benutzer erarbeitet.
- **Kommunikationsverzerrungen, die darauf beruhen, daß jeder nur seinem eigenen Text**[120] **folgt:** Die Kommunikationsereignisse des jeweiligen Kommunikationspartners werden unter subjektiven Gesichtspunkten interpretiert, die in keinem Zusammenhang mit den Absichten des Kommunikationspartners stehen. Der Benutzer hat beispielsweise die Absicht (Text), dem Entwickler die Schwierigkeiten und Probleme seiner Arbeitstätigkeit anhand von Sonderfällen darzustellen. Der Entwickler interpretiert dessen Äußerung unter der Perspektive (»nach dem Text«) »Ermittlung der Kernanforderungen des zukünftigen Softwaresystems«. Damit erhält der Ausdruck des Benutzers gewissermaßen einen ihm fremden »Kontext«. Das kontextuale Vergewissern (Feedback) über **Absichten und Interpretationshintergründe des Kommunikationspartners** ist eine Möglichkeit, die Interpunktionen der Kommunikation auf einen gemeinsamen Verstehenshorizont zurückzuführen.
- **Kommunikationsverzerrungen, die darauf beruhen, daß Kommunikationsereignisse nur als Assoziationsanlässe dienen:** Unklarheiten bezüglich des inhaltlichen Ziels und seiner thematischen Vorstrukturierung führen dazu, daß die Interpunktionen den Kommunikationsablauf nach dem Zufallsprinzip strukturieren. Ähnlich wie die Tagesordnung eine Gruppendiskussion mitgestaltet, benötigt jede sachliche Kommunikation strukturierende Elemente. Ob diese Elemente die Form eines Interviewleitfadens, eines all-

[120] Wir verwenden hier ausdrücklich den Begriff Text, da er sowohl auf den Begriff »Kontext« verweist, als auch deutlich macht, daß diesem Text nicht nur subjektiver Eigensinn zukommt, sondern auch intersubjektive Bedeutung zugrunde liegt.

gemeinen Diskussionsziels oder einer differenzierten Tagesordnung haben, kommt auf das spezifische Kommunikationsziel an. Für den Kommunikationsprozeß zwischen Entwicklern und Benutzern ist es u. E. wichtig, jeweils die folgenden Dimensionen der Kommunikation zu klären:

- Welche Funktion hat die Kommunikation im Entwicklungsprozeß? Gehen von den Kommunikationsergebnissen Handlungsimpulse aus? Sind diese Handlungsfolgen reversibel? Die Klärung dieser Fragen stellt die **Transparenz der Bedeutung des jeweiligen kommunikativen Handelns** sicher. Gleichzeitig entscheidet sie über die Form der Dokumentation des Kommunikationsablaufs.
- Wie offen, beziehungsweise wie geschlossen sollte die Kommunikation jeweils sein? Gibt es ein konkretes Kommunikationsziel, oder geht es nur um einen prinzipiellen Gedankenaustausch? Die konsensuale Klärung der **Balance zwischen Offenheit und Strukturiertheit** muß vor der eigentlichen Kommunikation stattfinden.

Auch für das Axiom der Interpunktion können Regeln der Kommunikation formuliert werden[121]:

1. »**Höre dem Sprecher gut zu** und benutze die Redezeit des anderen nicht nur dazu, darüber nachzudenken, was du als nächstes sagst! (Wenn du eine längere Bedenkzeit brauchst, fordere eine Pause ein!)«
2. »**Gehe auf die Bemerkungen des Redenden ein!** (Wenn du glaubst, diese gehen am Thema vorbei, so begründe dies, aber gehe nicht einfach über sie hinweg!)«
3. »**Wiederhole oder fasse Diskussionsbeiträge zusammen**, um ein Feedback zu erhalten, ob du alles richtig verstanden hast!«
4. »**Gestatte dem Sprecher, Pausen zu machen und ermuntere ihn zum Weitersprechen!** (Ausredenlassen heißt nicht zu warten, bis der Kommunikationspartner Luft holt, sondern bis er das, was er sagen will, gesagt hat!)«
5. »**Fasse dich kurz!** (Dein Gegenüber kann nur auf dich eingehen, wenn du deinen Redebeitrag auf wenige Punkte konzentrierst!)«
6. »**Kontakt kommt vor Kooperation!** (Auch funktionale, strukturierte Kommunikation bedarf eines offenen Auftakts)«

[121] Die im folgenden als hervorgehoben markierten Regeln sind entnommen aus: [Fleischer 1990, S. 93ff.].

Diese Regeln lösen zwar nicht alle Probleme der Interpunktion von Kommunikationsabläufen, aber sie bieten doch Ansatzpunkte, die notwendige Interpunktion der Kommunikation zum aktiven und bewußten Bestandteil des eigenen kommunikativen Handelns zu machen. Da die Reflexion und bewußte Setzung von Interpunktionen dezidiert Sprachkompetenz, Rollendistanz und Empathie erfordert und fördert, ist die Erarbeitung und bewußte Auseinandersetzung mit den oben genannten Regeln ein weiteres didaktisches Mittel zum Einüben sozialer Kompetenzen. Um diese Regeln adäquat einsetzen zu können, muß der Lehrende seine Aufmerksamkeit auf die Interpunktion von Kommunikationsabläufen richten.

Das Axiom der Unmöglichkeit nicht zu kommunizieren, das Axiom der digitalen und analogen Modalitäten der Kommunikation und ihr Verhältnis zu sozialen Kompetenzen

Da sich die Entwickler-Benutzer-Kommunikation fast ausschließlich durch digitale Modalitäten bewußter Kommunikation bestimmen läßt, sind diese Axiome von Watzlawick für unsere Betrachtungen nur am Rande von Interesse. Erst wenn Differenzen oder Übersetzungsfehler zwischen analoger und digitaler Kommunikation den Verständigungsprozeß von Entwicklern und Benutzern gefährden, werden diese Axiome für unseren Problembereich wichtig. Dies läßt sich am deutlichsten anhand des Beispiels personaler Achtung verdeutlichen: Analoge Kommunikation transportiert in erster Linie Nachrichten, die einen ausgeprägten Beziehungsaspekt haben. Das heißt aber auch: **Personale Achtung**, die eine Grundvoraussetzung der Benutzer-Entwickler-Kommunikation darstellt, **wird zum größten Teil analog kommuniziert.** Regeln der analogen Kommunikation, die personale Achtung erzeugen, können aber allein schon deshalb nicht aufgestellt werden, weil personale Achtung der Kommunikation vorausgehen muß. Andererseits kann es durchaus sein, daß Kommunikationsteilnehmer **analoge Kommunikation fehlerhaft in digitale Kommunikation übersetzen**: Der Benutzer arbeitet zwar konstruktiv mit, signalisiert aber nonverbal eine ablehnende Haltung, die der Entwickler auf seine Person, beziehungsweise auf seine Vorschläge bezieht. Tatsächlich will der Benutzer nur seine allgemeine Arbeitsüberlastung und die Aufforderung, darauf Rücksicht zu nehmen, kommunizieren, die durch die Projektarbeit noch zunimmt. **Als Regel der Kommunikation in funktionalen Arbeitsbezügen kann gelten, daß analoge Kommunikation dann in digitale übersetzt und geklärt werden sollte, wenn sie zur Störung der aufgabenbezogenen Kommunikation wird.**

Das Axiom, daß man nicht nicht kommunizieren kann, verweist auf die Vielschichtigkeit menschlicher Kommunikation. Insbesondere ist hier auf die **komplexe Mikropolitik der Macht-, Hierarchie- und Zuständigkeitsbeziehungen**

in Unternehmen hinzuweisen, die dazu führen, daß das Verhalten des Entwicklers zahlreiche, nicht intendierte kommunikative Nebenfolgen zeitigt. **Als Regel kann hier nur formuliert werden, daß die Entwickler die gegebenen mikropolitischen Strukturen in ihrem Verhalten berücksichtigen und auf projektspezifische Kommunikationsbeziehungen insistieren sollten** (z.B.: Transparenz für alle, indem der aktuelle Projektstand an einer Projektwand »öffentlich« dokumentiert wird).

Zusammenfassung

Kommunikationsregeln können zur Einübung und Förderung sozialer Kompetenzen didaktisch eingesetzt werden. Wichtig erschien es uns, von den oftmals defizitären impliziten und expliziten Regeln der Kommunikation auszugehen, wie wir sie oftmals im Unterrichtssystem vorfinden. Dabei unterstellen wir, daß diese Regeln auch soziale Kompetenzen unterminieren, die gleichermaßen für offene Formen des Unterrichtens (Rollenspiele, Projektarbeit, Gruppendiskussion) wie für die berufliche Handlungskompetenz notwendig sind. Soziale Kompetenzen können nach unserer Auffassung nicht einfach durch neue Unterrichtsformen hergestellt werden. Sie müssen vielmehr durch die Erarbeitung und Reflexion von Kommunikationsregeln in evidenten Lernsituationen erarbeitet werden. Die Kongruenz von Kommunikationsregeln und sozialen Kompetenzen sehen wir durch das verknüpfende Element der pragmatischen Axiome Watzlawicks gegeben[122].

Die Reflexion und Erarbeitung von Kommunikationsregeln in evidenten Lernsituationen bildet das Zentrum des didaktischen Handelns, welches sich die Einübung und Förderung sozialer Kompetenzen zum Ziel setzt. Erst nachdem wir dieses Zentrum bestimmt haben, können wir uns nun den äußeren Unterrichtsformen zuwenden, die diese Lernsituationen wahrscheinlicher machen.[123]

---

[122] Die Ableitung sozialer Kompetenzen aus den Erfordernissen menschlicher Kommunikation geht auf Habermas zurück. Habermas spricht anstelle von sozialen Kompetenzen auch von kommunikativen Kompetenzen [vgl. Habermas 1971]. Wir haben uns hier allerdings auf Watzlawick bezogen, da sich mit seinen Axiomen ein evidenter und auch praktisch nutzbarer Zusammenhang zwischen Kommunikationsregeln und sozialen Kompetenzen herstellen läßt.

[123] Der weitverbreitete Glaube, Projektarbeit und Rollenspiele würden schon per se soziale Kompetenzen fördern, wird allein schon durch die pädagogischen Alltagserfahrungen widerlegt, die vielmehr zeigen, daß Gruppenarbeit und Projektarbeit gerade am Mangel an sozialen Kompetenzen scheitert. Die damit einhergehenden negativen Erfahrungen der Schüler mit der Projekt- und Gruppenarbeit haben oftmals sogar negative Folgen für ihre Einstellungen zur kollektiven Arbeit.

5.2 Unterrichtsformen und soziales Lernen

»Soziales Lernen findet immer und überall statt.« Diese Aussage, deren Allgemeingültigkeit kaum jemand bezweifeln wird, kann in einer Situation direkter Kommunikation mehrere Nachrichten übermitteln. Erstens: Sie kann sachinhaltlich bedeuten, daß der organisierten Bildung kein privilegierter Stellenwert bei der Vermittlung von sozialen Kompetenzen zukommt, daß diese vielmehr in der Familie und in den peer-groups vermittelt werden. Zweitens: Sie kann den appellativen Charakter haben, bei jeder didaktischen Handlung auf die Implikationen für das soziale Lernen zu achten. Drittens: Sie kann die Selbstoffenbarung eines Pädagogen sein, der behauptet, daß er durch die Art und Weise seines kommunikativen Handelns permanent soziale Kompetenzen fördert. Viertens: Sie kann sogar einen Beziehungsaspekt ausdrücken, wenn sie bedeutet, daß man als Lehrer sich diesbezüglich von anderen Pädagogen nichts sagen läßt.

Die spekulative Reflexion über den potentiellen nachrichtlichen Aspekt dieser Aussage erfolgte hier keineswegs allein nochmals, um den praktischen Gehalt des Vier-Aspekte-Theorems Schulz von Thuns aufzuzeigen, sondern vielmehr, um auf einen Klärungsbedarf aufmerksam zu machen. Dieser Klärungsbedarf erstreckt sich auf folgende Fragen:

- Warum ist die systematische Förderung sozialer Kompetenzen durch bestimmte Unterrichtsformen notwendig, obwohl immer und überall soziales Lernen stattfindet? Wie sehen diese Unterrichtsformen aus?
- Wie können »evidente Lernsituationen« der sozialen Kompetenzen bei der Planung von Unterricht antizipiert werden?
- Wie können unterschiedliche Unterrichtsformen in die Ausbildung integriert werden, ohne daß es zu Widersprüchen im Lernen sozialer Kompetenzen kommt?
- Welche Kooperationsformen zwischen den Lehrenden unterstützen das Lehren sozialer Kompetenzen in kooperativen Unterrichtsformen?

Im folgenden werden wir uns der Diskussion dieser Fragen im Hinblick auf ihre Bedeutung für das praktische didaktische Handeln zuwenden.

5.2.1 Kooperative Unterrichtsformen zur systematischen Förderung sozialer Kompetenzen - Unterrichtsform und Arbeitsprozeß

Kooperative Unterrichtsformen, wie Gruppenarbeit, Rollenspiele und Projektarbeit, sind Voraussetzungen zur systematischen Förderung sozialer Kompetenzen im Unterricht. Kooperative Unterrichtsformen können bezüglich der Förderung

sozialer Kompetenzen gewissermaßen als didaktische Notwendigkeiten angesehen werden. Diese Notwendigkeit liegt darin begründet, daß in kooperativen Unterrichtsformen soziale Kompetenzen für die Bewältigung der Lernaufgabe **funktional** sind. Diese Funktionalität sozialer Kompetenzen beinhaltet einige didaktische Probleme, deren Nichtbeachtung zu unangemessenem didaktischen Handeln führen kann:

1. Die Funktionalität garantiert nicht, daß sich soziale Kompetenzen quasi als Nebenprodukt automatisch beim aufgabenorientierten Arbeiten in der Gruppe einstellen. Vielmehr ist davon auszugehen, daß die Notwendigkeit sozialer Kompetenzen auch soziale Lernhemmungen beim Lernenden generieren kann, die nur durch spontane und geplante Lernhilfen bewältigt werden können.
2. Die Funktionalität der sozialen Kompetenzen muß explizit auf den realen Arbeitsprozeß bezogen werden. Ihre Funktionalität darf nicht auf die konkrete Unterrichtsaufgabe beschränkt bleiben.

Da wir den ersten Punkt weiter unten unter dem Stichwort der Antizipation »evidenter Lernsituationen« näher betrachten werden, wollen wir uns vorerst nur dem Verhältnis zwischen kooperativen Lernaufgaben und dem Arbeitsprozeß zuwenden.

Kooperative Unterrichtsformen, insbesondere die Projektarbeit, aber auch das Rollenspiel, sind dem realen Arbeitsprozeß ähnlicher als die klassische Unterrichtsform des Frontalunterrichts. **Sowohl der Arbeitsprozeß als auch der Projektunterricht sind kooperativ und produktorientiert.** Gerade aber diese erhöhte Ähnlichkeit wird zu einem didaktischen Problem, das behutsames Vorgehen erfordert. Daß der Frontalunterricht nichts mit dem realen Arbeitsprozeß zu tun hat, weiß der Lernende so gut wie der Lehrende. Kooperative Projektarbeit, die sich beispielsweise die Erstellung eines Software-Produkts zur Aufgabe macht, erweckt dagegen den Anschein der Identität von Projektarbeit und realem Arbeitsprozeß. Diese Identität ist allerdings weder bezüglich der aufgabenbezogenen noch bezüglich der sozialen Projektkomplexität gegeben. Das didaktische Bemühen, über Projektarbeit den realen Arbeitsprozeß ganzheitlich einzufangen, ist damit prinzipiell zum Scheitern verurteilt. Einziges Resultat eines solchen Vorgehens wäre die Diffusion der Lernziele, die auch nicht mehr durch spontanes didaktisches Handeln sinnvoll bewältigbar wäre. **Die Konzeption kooperativer Unterrichtsformen stellt also keineswegs aus sich selbst heraus ein befriedigendes Verhältnis zum realen Arbeitsprozeß her.** Vielmehr müssen sich auch kooperative Unterrichtsformen auf einzelne Aspekte des analytisch zerlegten Arbeitsprozesses beziehen. Gleichzeitig müssen die aus

diesen Aspekten hervorgehenden Lernziele wieder didaktisch auf den realen Arbeitsprozeß zurückbezogen werden. Aus diesen grundsätzlichen Überlegungen können erste Leitlinien für die Konzeption von Lernaufgaben in kooperativen Unterrichtsformen abgeleitet werden:

- Die Lernaufgaben, die im didaktisch ausbalancierten Verhältnis vom realen Arbeitsprozeß zu den angenommenen Lernfähigkeiten der Lernenden bestimmt werden, bilden in der Regel[124] den Bezugspunkt kooperativer Unterrichtsformen: Die **Subjektorientierung der Didaktik** gilt als die primäre didaktische Handlungsorientierung bei der Gestaltung der Projektaufgabe, der Bezug zum realen Arbeitsprozeß ist stets nur ein über die berufliche Handlungskompetenz vermittelter. Die Fokussierung kooperativer Unterrichtsformen auf die Lernaufgabe verhindert deren Diffusion. Daraus folgt als weitere Leitlinie der Gestaltung kooperativer Unterrichtsformen:
- Die konzeptionelle Orientierung der Projektaufgabe oder der Rollenspielvorgaben ist **prozeßorientiert**. Produkte sind nie der Ausgangspunkt zur Konzeption von kooperativem Unterricht, sondern nur ihr Mittel. Zum Beispiel: Die Erstellung eines Pflichtenhefts kann nicht zum sinnvollen Ausgangspunkt eines Projekts gemacht werden. Vielmehr müssen die Probleme des Übergangs von konkretem Arbeitstätigkeitswissen der Benutzer in abstraktes Arbeitssystemwissen der Benutzer/Entwickler als nur kommunikativ lösbares Transformationsproblem zum Gegenstand der Lernaufgabe werden. Daß hierbei die Erstellung eines Pflichtenhefts auch eine didaktische Rolle spielen kann, soll hier nicht bestritten werden. Nur: Als Ausgangspunkt der Konzeption des Unterrichtsprojekts ist das Pflichtenheft ungeeignet.
- Da auch kooperative Unterrichtsformen die Trennung zwischen Unterricht und Arbeitsprozeß nicht aufheben können, muß der Bezug von der Lösung von Lernaufgaben und der Bewältigung von Aufgaben im realen Arbeitsprozeß **aktiv** didaktisch hergestellt werden. Raum dazu bietet sowohl die Einführung des Unterrichtsprojekts als auch die Nachbereitung kooperativen Unterrichts.

---

[124] Eine Ausnahme von der Regel stellen Projekte dar, die dezidiert der Ermittlung von Lernhemmungen und Lernfähigkeiten dienen, sowie Projekte, die die Evaluation von Unterricht zum Ziel haben. Beide Projektformen werden nicht primär als Unterrichtsformen des Lernens eingesetzt und haben somit eine Sonderstellung. Da diese Projektformen aber am verbreitetsten sind, besteht die Gefahr, daß sie als verallgemeinerungsfähige Modelle der Projektarbeit angesehen werden. Damit geht aber die Chance verloren, Projekte konzeptionell an der Aneignung beruflicher Handlungskompetenz durch die Lernenden auszurichten.

Als stark vereinfachtes Vorgehensmodell zur Konzeption eines Unterrichtsprojekts ergibt sich daraus das abgebildete Modell, das einen Zyklus der Konzeption darstellt. Die **Zyklizität** ist notwendig, weil nur in einem iterativen Verfahren die drei Faktoren **Lernfähigkeiten, subjektbezogene berufliche Handlungskompetenz und funktionaler Arbeitsprozeß** vermittelt werden können.

Dem Begriff der beruflichen Handlungskompetenz kommt dabei zentrale Bedeutung zu, da er zwischen dem funktionalen Arbeitsprozeß und den Lernfähigkeiten der Lernenden vermittelt. Gleichzeitig liefert er auch eine Matrix zur Bestimmung der Lernaufgabe:

1. Welches Wissen und welcher Umgang mit Wissen sollen vermittelt und moderiert werden?
2. Welche sozialen Kompetenzen sind funktional notwendig, und wie können sie gefördert werden?
3. Welche berufliche Identität ist angemessen und soll in ihrer Entwicklung unterstützt werden?

Die systematische Förderung sozialer Kompetenzen durch die Konzeption kooperativen Unterrichts ist daher möglich, da nicht ein bestimmtes Produkt zum Ausgangspunkt eines Unterrichtsprojektes wird, sondern die Kompetenzen selbst diesen Ausgangspunkt bilden.

5.2.2 Die Antizipation »evidenter sozialer Lernsituationen« als Planungselement bei kooperativen Unterrichtsformen

Die Antizipation von »evidenten sozialen Lernsituationen« ist nicht nur wichtig, um **die situative didaktische Wahrnehmung des Lehrenden zu fokussieren**. Die Antizipation von »evidenten sozialen Lernsituationen« ist auch Voraussetzung für **die Bereitstellung von geplanten Lernhilfen**. »Evidente soziale Lernsituationen« liegen immer dann vor, wenn Lernhemmungen konstruktiv durch die Aneignung von Lernhilfen vom Lernenden bewältigt werden können. Ein großer Teil dieser sozialen Lernhemmungen ist nicht vorhersehbar und kann damit nicht Gegenstand der Planung von kooperativen Unterrichtsmomenten sein. Auf diese sozialen Lernhemmungen muß jeweils spontan reagiert werden.[125] Andererseits können viele Lernhemmungen auch mittels der Reflexion der Kompetenzerfordernisse der Lernaufgaben und der Reflexion der vorhan-

[125] Unsere Ausführungen zu den Axiomen der Kommunikation und den Kommunikationsregeln bieten die heuristische Hilfe für das spontane Bereitstellen von Lernhilfen zur Bewältigung sozialer Lernhemmungen.

denen Lernfähigkeiten der Lernenden antizipiert werden. Diese Antizipation ermöglicht wiederum das Bereitstellen von Formen der Lernhilfen, die nicht ad hoc realisierbar sind. Im folgenden werden wir einige dieser Lernhilfen und antizipierten Lernsituationen exemplarisch skizzieren:

Beispiel: Perspektivenverschränkung im Systementwicklungsprozeß - das Verhältnis von Konkretem und Abstraktem als Problem sozialer Kompetenzen

Die Lernaufgabe einer kooperativen Unterrichtsform in Form eines Rollenspiels besteht beispielsweise darin, die konkreten Arbeitstätigkeiten der Benutzer in abstrakte Beschreibungen des Arbeitsprozesses zu transformieren, um eine operationalisierbare Basis für den ersten Zyklus des Systementwicklungsprozesses zu schaffen. Da diese Beschreibungen des Arbeitsprozesses auch die Grundlage weiterer Benutzer-Entwickler-Kommunikationen bilden sollen, werden als Anforderungen für diese Beschreibungen folgende Kriterien vereinbart:

1. Das Modell soll konsensual entwickelt werden.
2. Das Modell soll von allen Beteiligten tendenziell gleichermaßen verstanden werden.

Die für die Bearbeitung der Lernaufgabe erforderlichen kommunikativen Kompetenzen sind sowohl als Kommunikationsregeln als auch als konkrete Methoden (Interviewmethoden, Erstellen eines Glossars, Illustrations- Repräsentationsmethoden u.ä.) den Lernenden bekannt. Ebenso ist das Modell des problematischen Übergangs vom konkreten zum abstrakten Arbeitswissen schon theoretisch durchgearbeitet worden. Die Gefahr, daß die Benutzerrollen nicht durchgehalten werden, wird durch ausführliche Rollentexte und durch die Mitarbeit des Lehrenden bei der Benutzergruppe minimiert.

Als mögliche Lernhemmungen antizipiert der Lehrende z.B.:

- Die bekannten konkreten Methoden (Interviewmethoden usw.) werden nicht situationsangemessen angewendet, sondern oftmals nur schematisch.
- Konsens und Verstehen werden durch die Kommunikationsregeln zwar tendenziell hergestellt, aber diese werden kaum eingesetzt, um Konsens und Verstehen zu evaluieren. Es setzt sich leicht das Prinzip des »Informationsanzapfens« durch: Die Information, die man dem Anwender entlockt hat, »hat man« und gibt sie nicht so schnell wieder her oder auf.
- Das Transformationsmodell, das die Transformation vom konkreten Arbeitswissen des Benutzers zum abstrakten Arbeitssystemwissen beschreibt, wird zur »Objektivierung« des sozialen Prozesses eingesetzt. Im sozialen

Prozeß ist Abstraktes und Konkretes aber immer nur relativ und nicht eindeutig.

Für diese antizipierten möglichen Lernhemmungen kann der Lehrende geplante Lernhilfen entwickeln, die er systematisch im Unterricht einsetzt. Solche Lernhilfen können, auf die oben grob skizzierten Lernhemmungen bezogen, folgende sein:

- Nach einer sequentiellen Unterbrechung des Rollenspiels werden die problematischen Anwendungen von Methoden der Wissensakquisition thematisiert und systematisiert. Der Lehrer stellt als Lernhilfe beispielsweise eine Typologie nicht adäquater Anwendungen von Methoden vor, die in einer gemeinsamen Diskussion ergänzt wird. Diese Typologie dient einer Gruppe von Lernenden, die nicht direkt im Rollenspiel beteiligt sind, als Beobachtungsraster.
- Der Lehrende und die Lernenden haben das Recht, das Rollenspiel in vereinbarten Zeitabständen kurz zu unterbrechen, um Kommunikationsprobleme metakommunikativ problematisieren zu können. Eine vorzügliche Lernhilfe stellen Videoselbstkonfrontationen[126] dar (da das unbefangene Agieren vor laufender Videokamera eine gewisse Gewöhnung voraussetzt, sollte das Videographieren möglichst häufig eingesetzt werden). Darüber hinaus bekommt ein Teil der Lernenden, die nur als Beobachter am Rollenspiel teilnehmen, einen strukturierten Leittext, der ihre Aufmerksamkeit auf verzerrende Kommunikationsformen fokussiert.
- Der Lehrende bereitet eine themenzentrierte Diskussion vor, die das Problem der sozialen Vermitteltheit von den Begriffen »konkret« und »abstrakt« zum Gegenstand hat. Dazu stellt er zentrale allgemeine und spezifisch auf die Informatik bezogene Texte zusammen, die dieses Problem anschaulich darstellen. Eine Beobachtergruppe der Lernenden bekommt den Auftrag, eine Liste abstrakter und konkreter Aussagen bezüglich der Arbeit während des Rollenspiels zu erstellen. Diese Liste soll als weiterer Bezugspunkt in die Plenumsdiskussion eingehen.

Die hier in unserem Beispiel erwähnten Lernhilfen zeichnen sich dadurch aus, daß sie gleichermaßen **direkte und indirekte Lernhilfen** darstellen. In der Form von Texten und Instruktionen haben sie den Charakter direkter, z.T. sogar direktiver Lernhilfen. Gleichzeitig konstituieren sie allerdings die Beobachtergruppe

[126] Das Videographieren an sich kann allerdings nicht als Lernhilfe bezeichnet werden. Erst wenn die Videoselbstkonfrontation mit einer spezifischen Fragestellung verknüpft wird, sprechen wir von einer Lernhilfe.

als drei unterschiedliche Expertengruppen, die ihre jeweilige Perspektive untereinander und mit den Rollenspielern vermitteln müssen. Insofern diese Lernhilfen einen kommunikativen Vermittlungsprozeß in Gang setzen, der zur Bewältigung von Lernhemmungen, die aus einer verengten Problemperspektive resultieren, beiträgt, können diese Lernhilfen auch als indirekte Lernhilfen bezeichnet werden. **Insbesondere als indirekte Lernhilfen zielen sie auf die Förderung von sozialen Kompetenzen.** Im folgenden Abschnitt wollen wir uns Lernhilfen zuwenden, die nicht nur indirekt einen kommunikativen Vermittlungsprozeß zwischen »Experten« konstituieren, sondern Hilfen zur Bewältigung von Lernhemmungen in diesem Prozeß darstellen.

Voneinander Lernen: Lernhilfen zur Unterstützung kooperativen Lernens

Die Befähigung der Lernenden, voneinander zu lernen, das heißt, autonome Lernbeziehungen einzugehen, ist im Falle der IT-Ausbildung nicht nur aus prinzipiellen Gründen als pädagogisches Ziel anzusehen, da für die adäquate Entwicklung von Software die Herstellung einer Lernbeziehung zwischen Entwicklern und Benutzern absolut notwendig ist (vgl. Kapitel 4 des Rahmencurriculums »Moderne Software-Entwicklung«). Voraussetzung für autonome Lernbeziehungen, die nicht dem Rollenschema Lehrer-Lernende unterworfen sind, ist es, daß der Lehrende sich zurücknimmt. Dieses Zurücknehmen ist aber weder für den Lehrenden noch für die Lernenden einfach. Die »normalen« Interaktionserwartungen und -formen führen zu Lernhemmungen, die kooperatives Lernen zwischen den Lernenden leicht verhindern können. Die Bewältigung dieser Lernhemmungen kann u.E. nicht allein durch spontanes didaktisches Handeln gewährleistet werden. Vielmehr müssen geplante Lernhilfen die Grundlagen für kooperatives Lernen schaffen. Für unser Beispiel könnten diese geplanten Lernhilfen folgendermaßen aussehen:

- Neben den Lernhilfen, die die Wahrnehmung der Beobachtungsgruppen fokussieren und strukturieren, werden Lernhilfen bezüglich der Repräsentation der Ergebnisse zur Verfügung gestellt. Die Lernhilfen können z.B. die Form von Leitfragen haben (Welches Ergebnis ist uns am wichtigsten? Warum ist es auch für die Rollenspieler wichtig? Wie können wir das Ergebnis anschaulich darstellen? Auf welche Punkte sollen wir uns konzentrieren? An welchen Punkten wollen wir ein Feedback haben? Welche Fragen haben wir, die wir noch den Rollenspielern stellen wollen?...). Die Lernhilfen können aber auch die Form des Vorschlags einer konkreten Repräsentationstechnik annehmen.
- Wenn der Lehrende, aufgrund der Annahme der vorhandenen Lernfähigkeiten der Lernenden, eine Phase der Auswertung ohne seine persönliche

Teilnahme (oder nur moderierende Teilnahme) vorsieht, kann er für diese Phase Lernhilfen zur Verfügung stellen. Die Rollenspieler erhalten beispielsweise eine Lernhilfe für die Auswertung, die ebenfalls die Form von Leitfragen hat (welche Aussagen der Beobachtergruppe halten wir für die wichtigsten? Welche teilen wir, welchen stimmen wir nicht zu? Welche Bedeutung haben die Aussagen für unser Verhalten?...). Der Sinn dieser Lernhilfe kann noch dadurch verstärkt werden, daß sie mit einer konkreten Lernaufgabe verbunden wird (z.B.: die Rollenspieler müssen im Plenum berichten, was die Beobachtungsgruppe beobachtet hat, und wie sie dazu stehen).

Der Lehrende, der kooperatives Lernen durch Lernhilfen fördern will, wird bei der Planung des Unterrichts mit einer **Paradoxie seines pädagogischen Handelns** konfrontiert, die er für die Bereitstellung von Lernhilfen berücksichtigen muß: Im Prozeß des kooperativen Lernens, das zwischen den Lernenden stattfindet, entfaltet sich oftmals eine soziale Eigendynamik, die den kooperativen Lernprozeß gefährdet. Interaktionserwartungen, problematische Perspektiven, persönliche Konflikte, aber auch fehlendes Problemlösewissen können den kooperativen Lernprozeß blockieren. Werden diese Lernhemmungen allerdings durch das vorschnelle Bereitstellen von Lernhilfen beseitigt, wird das ursprüngliche Ziel, autonome Lernbeziehungen zu unterstützen, konterkariert. Die Balance zwischen Autonomie und Außensteuerung kooperativer Lernprozesse kann didaktisch nur durch die **Balance zwischen aufgabenorientierten Lernhilfen und lernprozeßorientierten Lernhilfen** hergestellt werden:

- Als aufgabenorientierte Lernhilfen bezeichnen wir Lernhilfen, die primär zur Bewältigung von Lernhemmungen dienen, die durch die spezifische Lernaufgabe hervorgerufen werden. Das Spektrum aufgabenbezogener Lernhilfen reicht vom Geben einer zur Lösung notwendigen, aber nicht erarbeitbaren Information (z.B. eine spezifische Methode) bis hin zur Bereitstellung von Fragen, deren Relevanz und Beantwortung durch die Lernenden selbst erschlossen werden muß.
- Als lernprozeßorientierte Lernhilfen bezeichnen wir Lernhilfen, die primär zur Bewältigung von Lernhemmungen dienen, die durch die soziale Eigendynamik kooperativer Lernprozesse hervorgerufen werden. Auch hier finden wir eine weite Spannbreite didaktischen Handelns. Sie reicht von der Einübung expliziter Kommunikationsregeln bis hin zum Bewußtmachen persönlicher Konflikte, lernhemmender Interaktionserwartungen und lernhemmender (vor-)beruflicher Identitäten bei den Lernenden.

Ohne hier ins Detail gehen zu können, was die Kenntnis der konkreten Lernenden und Lehrenden, der Ausbildungsorganisation und der konkreten Lernaufgabe gleichermaßen voraussetzen würde, kann für die Planung und Durchführung kooperativer Unterrichtsformen zusammenfassend resümiert werden: Kooperativer Unterricht stellt hohe didaktische Anforderungen sowohl an die Konzeption der Lernaufgabe, an die Antizipation »evidenter Lernsituationen« und an die Bereitstellung adäquater Lernhilfen, als auch an die didaktische Aufmerksamkeit während der Durchführung des kooperativen Unterrichts. Um diese Anforderungen bewältigen zu können, müssen Voraussetzungen in der Organisation der Ausbildung geschaffen werden, die den einzelnen Lehrenden entlasten:

1. Kooperative Unterrichtsformen müssen als verbindlicher Bestandteil in die Ausbildung integriert werden. Kooperativer Unterricht kann nicht jedesmal aufs neue als »nicht normaler« Unterricht gegen eingeschliffene Interaktionserwartungen durchgesetzt werden.
2. Kooperative Unterrichtsformen bedürfen der Kooperation in der Planung und Durchführung zwischen den Lehrenden. Erst durch diese Kooperation können kooperative Unterrichtsformen gemeinsam weiterentwickelt werden.

5.3 Zusammenfassung: Didaktisches Handeln und soziale Kompetenzen

Ausgangspunkt unserer Ausführungen zur didaktischen Vermittlung von sozialen Kompetenzen bildete die Überlegung, daß soziale Kompetenzen nur in der Form von **vorliegenden Kommunikationsereignissen** didaktische Handlungsmöglichkeiten bieten können. Vorliegende Kommunikationsereignisse haben nicht nur einen empirischen, logischen oder normativen Aussagewert, sondern sie werden auch maßgeblich von **Kommunikationsregeln** bestimmt, die sowohl den Sozialisationshintergrund der Kommunikationsteilnehmer als auch die Interaktionskultur des Systems Unterricht widerspiegeln. Die Bewußtmachung der vorhandenen Kommunikationsregeln kann als erster Schritt des didaktischen Handelns bezeichnet werden. Dadurch, daß bisher implizite Kommunikationsregeln metakommunikativ expliziert werden, werden sie kritisierbar. In einer Art von negativer Didaktik werden für das Kommunikationsziel **dysfunktionale Kommunikationsregeln** benannt und neue Kommunikationsregeln erarbeitet. Zur Erarbeitung adäquater Kommunikationsregeln, die gleichzeitig die sozialen Kompetenzen fördern, haben wir die **pragmatischen Axiome der Kommunikation** Watzlawicks herangezogen. Diese Axiome können gewissermaßen als Bindeglied zwischen sozialen Kompetenzen auf der einen Seite und Kommunikationsereignisse generierenden Kommunikationsregeln auf der anderen Seite angesehen werden. Indem wir die pragmatischen Axiome bis hin zu exemplari-

schen Kommunikationsregeln konkretisiert haben, konnten wir den **didaktischen Handlungsbereich »Soziale Kompetenzen«** präzise bestimmen. Insofern der Lehrende keine therapeutische Position einnehmen kann, behaupten wir, daß wir damit den didaktischen Handlungsbereich »Soziale Kompetenzen« auch vollständig beschrieben haben.

Nach dieser Bestimmung konkreten didaktischen Handelns, das sich auf die Förderung sozialer Kompetenzen bezieht, haben wir uns der **Konzeption von kooperativem Unterricht** zugewandt. Im kooperativen Unterricht ist die Lösung einer Lernaufgabe auch funktional mit dem Vorhandensein von sozialen Kompetenzen verbunden. Daß diese Kongruenz nicht automatisch die sozialen Kompetenzen fördert, haben wir explizit herausgearbeitet, um das didaktische Problembewußtsein zu steigern. Unsere Grundüberlegung ist es gewesen, daß sich die Konzeption kooperativen Unterrichts nur in einer **zyklischen Annäherung von Lernaufgaben und Lernsituationen**[127] adäquat bewerkstelligen läßt. In diesem zyklischen Prozeß gehen der funktionale Arbeitsprozeß, die subjektbezogene berufliche Handlungskompetenz und die Lernfähigkeiten der Lernenden als **auszubalancierende Faktoren** ein. Dieses zyklische Vorgehen führt zur Vermeidung einer verfehlten didaktischen Orientierung, die nur Produkte zum Ausgangspunkt der Konzeption kooperativen Unterrichts macht. Nachdem wir die Konzeption von kooperativem Unterricht als Annäherung von Lernsituation und Lernaufgabe bestimmt hatten, haben wir uns der **Antizipation evidenter sozialer Lernsituationen als Antizipation sozialer Lernhemmungen** zugewandt. Ziel dieser Antizipationen ist es, geplante Lernhilfen zur Verfügung zu stellen, die ein aktives, auf subjektiver Aneignung beruhendes Lernen bei den Lernenden fördern. Beispielhaft haben wir verschiedene Formen solcher Lernhilfen dargestellt, die sich wiederum unter den Gesichtspunkten **direkte/ indirekte und aufgabenorientierte/ lernprozeßorientierte Lernhilfen** klassifizieren lassen.

Am Schluß unserer Ausführungen haben wir noch auf zwei weitere Aspekte hingewiesen, die die Durchsetzung kooperativer Unterrichtsformen in der Ausbildung maßgeblich bestimmen, aber nicht in der Entscheidungsmacht einzelner Lehrender liegen. Der eine Aspekt betrifft die **Integration kooperativer Unter-**

---

[127] Hier wird deutlich, wie die Struktur der Planung des kooperativen Unterrichts zu einer Erweiterung und Korrektur klassisch lerntheoretisch ausgerichteter Unterrichtsplanungen führt. Die einfache Intentionalität (»was soll gelernt werden?«) wird zur komplexen Bestimmung von Lernaufgaben, die thematische Bestimmung des Lerngegenstandes wird zum komplexen Arrangement einer Lernsituation, die an Lernaufgaben konzeptionell zurückgekoppelt ist. Die Methodik wird nicht bezugslos festgelegt, sondern als Lernhilfen auf antizipierte Lernhemmungen zurückbezogen usw.

richtsformen als »Normalität« schulisch organisierter Ausbildung. Der andere Aspekt betrifft die **Kooperation zwischen den Lehrenden**, die erst eine differenzierte pädagogische Betreuung kooperativen Unterrichts ermöglicht.

Nach unserer Darstellung des didaktischen Umgangs mit Wissen und sozialen Kompetenzen wollen wir uns im folgenden mit didaktischem Handeln und seiner Bedeutung für die Entwicklung der beruflichen Identität beschäftigen. Wie wir sehen werden, beinhaltet didaktisches Identitätslernen immer den didaktischen Umgang mit Wissen und sozialen Kompetenzen.

6 Didaktisches Handeln und die Entwicklung der beruflichen Identität

Die Besonderheit des Identitätslernens besteht nicht im Aufschließen eines gänzlich neuen didaktischen Handlungsbereichs, sondern in einer spezifischen Perspektive des didaktischen Handelns. Diese didaktische Handlungsperspektive geht von zwei Grundannahmen aus:

1. Identitätslernen ist eine ständige Nebenfolge von Lernprozessen, die funktional vom Lehrenden zur Aneignung von Wissen und sozialen Kompetenzen initiiert werden. Identitätslernen ist der Anteil des Lernprozesses, in dem der Lernende das Gelernte gleichzeitig auf ein kohärentes Selbstbild und ein angenommenes Berufsbild bezieht. Die Qualität dieser subjektiv hergestellten Beziehung zwischen Gelerntem auf der einen Seite und dem Selbstbild und dem angenommenen Berufsbild auf der anderen Seite entscheidet darüber, welche Relevanz das Gelernte für die subjektive berufliche Handlungsfähigkeit erlangt. Von Identitätslernen kann dann gesprochen werden, wenn das Gelernte zu einer nachhaltigen Veränderung des Selbstbildes und des angenommenen Berufsbildes führt. Die »Dreiecksbeziehung« von Wissen und Fähigkeiten, Selbstbild und subjektiv angenommenem Berufsbild kann nur dann zufriedenstellend stabilisiert werden, wenn das auf Wissen und soziale Kompetenzen ausgerichtete didaktische Handeln eine kohärente Identität impliziert. Konkret: Was ich an Wissen und Fertigkeiten lerne, muß ich in irgendeiner Form mit meinem Selbstbild und meinen Vorstellungen von meiner beruflichen Rolle (meinen beruflichen Aufgaben) in Einklang bringen können.
2. Identitätslernen wird zum direkten »Gegenstand« didaktischen Handelns, wenn spezifisches Wissen und spezifische soziale Lernerfahrungen vermittelt werden, die das subjektive Verständnis beruflicher Aufgaben zum Ziel haben. Dies impliziert eine andere didaktische Fragestellung bezüglich der beruflichen Handlungskompetenz: Nicht die Frage, was der beruflich Handelnde können und wissen muß, um angemessen handeln zu können, steht

nun im Mittelpunkt, sondern die Frage, welche Einstellungen und Motivationen dieses Können und Wissen handlungsrelevant werden lassen. Direktes Identitätslernen heißt demnach, die Einstellungen und Motivationen der Lernenden dezidiert zum Bezugspunkt didaktischen Handelns zu machen.

Im folgenden werden wir beide Formen des Identitätslernens anhand von Beispielen erläutern. Dabei werden wir jeweils wieder von einer negativen Position ausgehen, um eine positive zu entwickeln.

1. Wir fragen also zuerst, welche Lernorganisation und welche Lernform die Entwicklung einer unangemessenen beruflichen Identität als Nebenfolge zeitigt.
2. Danach entwickeln wir selbst Vorschläge, wie diese negativen Formen überwunden und durch positive ersetzt werden können.
3. Genauso verfahren wir bezüglich der zweiten direkteren Art des Identitätslernens, indem wir zuerst aufzeigen, wie die Entwicklung einer unangemessenen Identität direkt unterstützt wird.
4. Im Anschluß daran bieten wir Vorschläge, wie die Entwicklung einer adäquaten Identität direkt unterstützt werden kann.

6.1 Negative Formen der Identitätsentwicklung als Nebenfolgen der Lernorganisation und der Lernformen - zur Hegemonie des Programmcodes in der Ausbildung

Als »negative Formen der Identitätsentwicklung« bezeichnen wir diejenigen Formen der Identitätsentwicklung, die das erforderliche **Problembewußtsein**, daß Software-Entwicklung nur als Problem der Arbeitsgestaltung angemessen zu bewältigen ist, nicht nur nicht fördern, sondern blockieren. Insbesondere verstehen wir unter diesen negativen Formen der Identitäten die Identität des »Hackers«, der sein Problembewußtsein allein auf den Programmcode bezieht, und den »Techniker«, der seine Ambitionen allein auf die Herstellung der immanenten Funktionalität des technischen Systems richtet.

Unter Lernorganisation verstehen wir in unserem Zusammenhang die **verzeitlichte Sachlogik des Ausbildungsgangs** und die **sachlogische Struktur des Fächerkanons**.

Als Lernformen betrachten wir hier die **Art der Aufgabenstellungen** im Unterricht, den **Lehrenden als »Lernmodell«** und die **Lernform des Praktikums**.

Nach dieser kurzen Skizzierung unserer Begrifflichkeiten können wir folgende negative Formen des Identitätslernen als Nebenfolgen der **Lernorganisation** der IT-Ausbildung feststellen:

1. Die verzeitlichte Sachlogik der Ausbildung ist weitestgehend am Programmcode ausgerichtet:

   Die Ausbildung beginnt mit dem Programmieren eines einfachen Algorithmus. Im Laufe der Ausbildung werden dann immer komplexere Programme kodiert. Der Programmcode bietet den roten Faden der Ausbildung, um den sich andere Fächer gruppieren. Selbst das Fach Software Engineering, das praktisch zur Korrektur dieser Ausrichtung auf den Programmcode eingeführt wurde, muß sich dem Primat des Programmierens fügen. Software Engineering wird, sofern es überhaupt eine relevante Bedeutung in der Ausbildung erlangt, hauptsächlich als Mittel angesehen, um zum Programmieren zu kommen. Software Engineering dient dann allein dazu, die Hindernisse auf dem Weg zur eigentlichen Arbeit, dem Programmieren, möglichst schnell und systematisch zu beseitigen. Dieses »falsche« Bewußtsein stellt sich nicht nur als ein bedauerliches »Mißverständnis« auf Seiten der Lernenden ein, sondern die Organisation der Ausbildung unterliegt ihm selbst: Erst wenn das voraussetzungslose Programmieren an seine Grenzen stößt, werden formale Methoden der Software-Entwicklung eingeführt, die das Programmieren auch dann noch möglich machen, wenn es, überspitzt formuliert, eigentlich gar nicht mehr möglich wäre.

   Die Präsentation der IT-Ausbildung als das Schreiben immer komplexerer Programmcodes führt zu einer **Hegemonie des Programmierens**, dem alle anderen Fächer untergeordnet sind. Ein guter Informatiker ist derjenige, der komplexe Programmcodes schreiben kann. Eine negative Form der Identitätsentwicklung ist eine logische Nebenfolge einer um den Programmcode gruppierten IT- Ausbildung.

2. **Die verzeitlichte Sachlogik der Ausbildung, die Ausrichtung am Programmcode, hegemoniert die Beziehungen zu anderen Fächern.**

   Ähnlich problematisch wie das Fach Software Engineering, wenn auch anders gelagert, stellt sich die **Unterwerfung des Faches Mathematik unter den Primat des Programmcodes** dar. Die angewandte Mathematik gilt oftmals als die - nicht zuletzt auch legitimierende - Fundierung der Programmierung in der formalen Logik. Die »Hilfswissenschaft Mathematik« dient der Lösung des Korrektheitsproblems der Programmierung, die sich als strikte Formalisierung der realen Welt versteht. Indem die Mathematik nun selbst als einzige in sich geschlossene »wahre« Formalisierung präsentiert

> wird, wird das immanente Korrektheitsproblem - als allgemeines Problem der Gültigkeit - mit der Wirklichkeit identifiziert. Dieser Umgang mit Mathematik impliziert eine Einstellung, die eine Ausrichtung der Software-Entwicklung auf das Ziel der Arbeitsgestaltung unterminiert: Programmierung = Mathematik (= logisch = wahr). Das Kernproblem der **Formalisierung lebendiger Arbeit** taucht so nicht mehr auf. Gleichzeitig wird die angewandte Mathematik zum Absolutum, dessen Grundlagen nicht mehr hinterfragt und begründet werden können. Obgleich das Unterrichtsfach Mathematik im Rahmen der IT-Ausbildung kaum metamathematisch fundiert werden kann, so ist u.E. zumindest eine wissenschaftsphilosophische Reflexion mathematischer Grundbegriffe nötig und möglich.[128]
>
> Andere Fächer, wie zum Beispiel das Fach »Informatik und Gesellschaft«, werden durch die Hegemonie des Programmierens vollständig an den Rand gedrängt. Software-Ergonomie, rechtliche Fragestellungen und gesellschaftlich politische Probleme der Informatik erscheinen nur als äußere Hülle der Tätigkeit des Programmierens. Als Hülle sind sie dem Programmieren als Kerntätigkeit zeitlich vor- oder nachgelagert. Mit der »eigentlichen« beruflichen Tätigkeit, dem Programmieren, haben sie daher auch in der Ausbildung wenig zu tun.

Zusammenfassend kann gesagt werden: Die Ausrichtung der Ausbildung auf den Programmcode zeigt sich zuerst darin, daß das Programmieren den einzigen durchgängigen roten Faden der Ausbildung darstellt. Andere Inhalte haben oftmals die implizite Funktion sicherzustellen, daß dieser Faden bei der Steigerung der Komplexität nicht reißt. Diese verzeitlichte Sachlogik der Ausbildung führt zu einer Hegemonie des Programmierens im Fächerkanon. Diese Hegemonie zeigt sich darin, daß bestimmten Fächern, wie z.B. der Mathematik, nur wegen ihrer Funktion für das Programmieren Bedeutung beigemessen wird; andere Fächer, denen diese Bedeutung nicht zugesprochen wird, werden dagegen vollständig an den Rand gedrängt. Indem die Lehrenden diese **Hegemonie zur Motivation der Lernenden nutzen**, unterwerfen sie sich selbst dem Primat der Programmierung, auch wenn sie andere Sichtweisen in die Ausbildung hineintragen wollen.

---

[128] Die Mathematik erfährt in der IT-Ausbildung gleichermaßen eine Unter- wie eine Überbewertung: Als absoluter Bezugspunkt wird sie überbewertet, als nicht reflexionsfähig wird sie unterbewertet (Reflexionsbedürftig ist u.E. beispielsweise der Funktionsbegriff, das Verhältnis von Anschauung und mathematischen Begriffen, die Differenz von mathematischer Komplexität und der Komplexität von Computerprogrammen usw. Vgl. zur Einführung in metamathematische Fragestellungen:[Otte 1994]).

Wie sich diese Hegemonie auch in den einzelnen Lernformen umsetzt, wollen wir hier kurz skizzieren:

1. **Die Aufgabenstellungen werden um Programmieraufgaben gruppiert.** Dies gilt auch dann, wenn das Lernziel der Aufgabe nichts mit der Konstruktion eines Programmcodes zu tun hat; z.B. hat das Analysieren, Modellieren und aspektgebundene Darstellen von Arbeitssystemen **an sich** keinerlei Bezug zum Konstruieren als Programmieren. Die Fixierung auf das Erstellen eines Programmcodes verhindert sogar das angemessene Verstehen von Systemen. Wendt empfiehlt daher: "Solange man in den ersten Semestern der Informatikausbildung nicht auf das Programmieren verzichtet zugunsten des Modellierens und Darstellens informationeller Strukturen, kann aus den Studenten nichts anderes werden als fortgeschrittene Programmierer" [Wendt 1993, S. 38][129]. Aber selbst wenn viele Lehrende die Einschätzung Wendts prinzipiell teilen, so verzichten sie kaum darauf, jedes Wissen auf seine funktionale Bedeutung für das Programmieren rückzubeziehen. Indem somit die Orientierung am Programmcode didaktisch zur Motivierung der Lernenden eingesetzt wird, wird als praktische Nebenfolge die Hegemonie des Programmcodes weiter verstärkt. Die Aussage von Lehrenden, daß sie nur Wissen und Fähigkeiten vermitteln können, deren Nutzen für das Programmieren den Lernenden einsichtig ist, wird so zur **selffulfilling prophecy**.

2. Der Lehrende verkörpert selbst ein **Identitätsmodell**, das der Identität des »Technikers« entspricht. Identitätslernen findet häufig auch als Modellernen statt. Wenn wir hier dezidiert schreiben, daß der IT-Lehrende oftmals das Identitätsmodell »Techniker« verkörpert, und nicht, daß der Lehrende die Identität eines »Technikers« hat, so tun wir dies aus folgendem Grund: Der unterrichtende Lehrende steht vor dem Dilemma, daß er sich weder als Praktiker auf die fachlichen Inhalte beziehen kann, weil er außerhalb der IT-Praxis agiert, noch daß er sich selbst als Lehrer verstehen kann, weil er keinen ideellen und/oder ausbildungsbedingten Bezug zur Pädagogik erworben hat. Die Position des »exemplarischen Technikers» spiegelt somit eine verzerrte Erscheinung des »exemplarischen Intellektuellen« wider. Das Fatale in der Position des »exemplarischen Technikers« ist allerdings, daß er sein getrenntes Verhältnis zur Praxis nicht als eine aktiv intellektuell zu

---

[129] Diese Forderung Wendts teilen wir allerdings nur bedingt. Der gänzliche Verzicht auf das Programmieren hieße u.E., das Kind mit dem Bade auszuschütten. Abgesehen davon, daß ein »Programmierverbot« gegen die Ausbildungsinteressen der Lernenden nicht durchsetzbar ist, halten wir an unserem Vorschlag fest, Programmiersprachen von Anfang an in eine umfassende Fachlichkeit »Formalisierung« einzubetten.

bewältigende Aufgabe auffaßt, sondern die Haltung des »so-tun-als-ob« bezüglich der Praxis einnimmt. Ebensowenig ist es dem »exemplarischen Techniker« möglich, das Problem der Vermittlung der beruflichen Handlungskompetenz anders als technisch zu erfassen. Daß er sich trotz dieser Widersprüche im Ausbildungssystem reproduzieren kann, liegt daran, daß er aufs engste mit der Hegemonie des Programmcodes verbunden ist.

Wir halten fest: Negative Formen der Identitätsentwicklung sind auch Resultat des Modellernens. Als Modell dient die Positionseinnahme des Lehrenden als »exemplarischer Techniker«, der sich letztendlich auf die Hegemonie des Programmcodes stützen kann.

3. Die Lernform des Praktikums kann ebenfalls zur Verstärkung negativer Formen der Identitätsentwicklung beitragen. Diese Behauptung erscheint auf den ersten Blick widersprüchlich, da ja gerade die praktischen Erfahrungen die Grundlegung einer adäquaten beruflichen Identität fördern sollten. Dieser Widerspruch löst sich allerdings auf, wenn die Arbeitstätigkeiten während des Praktikums näher betrachtet werden. Da der Praktikant praktisch arbeiten soll, aber gleichzeitig nicht ganze Projekte begleiten kann, weil dies sowohl den zeitlichen Rahmen des Praktikums sprengen würde, als auch aus der Sicht der Projektleitung nicht verantwortbar wäre, bekommt er nur spezifische Tätigkeiten übertragen. Diese spezifischen Tätigkeiten sind meist wiederum auf abgekapselte, durch konkrete Zielbestimmungen eindeutig evaluierbare Programmieraufgaben bestimmt. Diese Form der Produktorientierung vieler Praktikas führt gewissermaßen automatisch zu einer Fixierung auf die Programmiertätigkeit. Die Organisation des Praktikums beinhaltet somit einen pädagogischen Zielkonflikt, der nur vermittelt bewältigbar ist: Der Praktikant soll im Praktikum nützliche Arbeit leisten und er soll im Praktikum berufliche Handlungskompetenz möglichst umfassend realisieren. Beides ist in der Realität oftmals unvereinbar. Die Lösung dieses Problems erfordert u.E. eine Vor- und Nachbereitung des Praktikums, die die Reflexion der eigenen Tätigkeit in Bezug zum Projektkontext ermöglicht. Ansonsten unterstützt auch das Praktikum eine negative Form der Identitätsentwicklung, indem es zu einer weiteren Stärkung der Hegemonie des Programmiercodes beiträgt. Welche Richtlinien eine adäquate Vor- und Nachbereitung zu beachten haben, werden wir weiter unten näher skizzieren.

Zusammenfassend können wir feststellen: Identitätslernen ist immer auch eine Nebenfolge der Lernformen. Aufgabenstellungen, das Identitätsmodell des Lehrenden und die Form und der Umgang mit dem Praktikum implizieren die Entwicklung einer bestimmten Identität. Um so mehr sich diese Lernformen an der

Hegemonie des Programmiercodes ausrichten, desto eher ist die Entwicklung einer beruflichen Identität wahrscheinlich, die eine adäquate Auseinandersetzung mit Problemen der Arbeitsgestaltung blockiert. Im folgenden werden wir Vorschläge machen, wie diese negativen Nebenfolgen der Lernorganisation und der Lernformen vermieden, beziehungsweise positiv gewendet werden können.

6.2 Lernorganisation und Lernformen als Ausgangspunkt einer adäquaten beruflichen Identität

Programmatisch kann formuliert werden: Die Lernorganisation als verzeitlichte sachlogische Struktur der Ausbildung und der Fächerintegration muß ebenso wie die unterschiedlichen Lernformen (Aufgaben aneignen, Modellernen, Erfahrungslernen) gegen die hegemonialen Tendenzen des Programmiercodes gerichtet sein. Diese hegemonialen Tendenzen des Programmiercodes rühren nicht nur daher, daß das Erlernen und Einüben von Programmiersprachen einen großen Teil der Ausbildungszeit kontinuierlich beanspruchen, sondern auch daher, daß die vorberuflichen Vorstellungen hauptsächlich um die Programmiertätigkeit kreisen. Die Fähigkeit, programmieren zu können, unterscheidet die IT-Fachkraft von allen anderen Berufen. Als **Differenzbegriff** stiftet diese Fähigkeit mithin Identität.

Die Konzeption einer Ausbildung, die gegen das Programmieren gerichtet wäre, hieße allerdings, das Kind mit dem Bade auszuschütten und unsere Formulierung, Ausbildung gegen die Hegemonie des Programmiercodes zu konzipieren, reichlich mißzuverstehen. Es kann vielmehr »nur« darum gehen, **die Programmiertätigkeit als Gestaltungstätigkeit ganzheitlich zu betrachten** und sie nicht unter ihren Elementen, den Codes, zu subsumieren. Der Informatiker Rolf vergleicht die gestaltende Tätigkeit der IT-Fachkräfte mit der Tätigkeit von Architekten. Die Fixierung auf den Programmcode würde einem Architekten gleichen, der seine Tätigkeit allein in der Umsetzung der Baustatik sehen würde. Eine Architektenausbildung, die jede Gestaltungsaufgabe mit der Aufgabe, einen detaillierten Bauplan zu entwerfen, verbinden würde, ist unvorstellbar. Genauso wie sich zukünftige Architekten ihre Tätigkeit einerseits durch bauingenieurliches Wissen, andererseits durch die Betrachtung gestalteter Architektur erschließen, sollte u.E. die IT-Ausbildung organisiert sein. Neben das Erlernen des Schreibens von Programmcodes tritt **die Betrachtung fertiger Programme als gestaltete Wirklichkeit**. Ebenso wie Gebäude etwas anderes sind als baustatische Berechnungen und aufeinander geschichtete Steine, sind Computerprogramme in der Arbeitswirklichkeit etwas anderes als aneinandergereihte Programmcodes. Der Hegemonie des Programmcodes ist u.E. eine Perspektive

gegenüberzustellen, die Programmieren als Gestaltung faßt, die nicht durch die Analyse des Programmcodes verstehbar ist.

Die Gestaltungsperspektive, die die Gestaltungsdimension als Entscheidungen eines sozialen Gestaltungsprozesses faßt, ist Voraussetzung für die Entwicklung einer angemessenen beruflichen Identität. Diese Gestaltungsperspektive ist **nicht** gleichzusetzen mit einer top-down-Strategie der Software-Entwicklung, die am »Ableitungsgedanken« festhält und somit impliziert, daß sie - gewissermaßen auf halbem Wege - sich mit der bottom-up-Strategie trifft. Die Gestaltungsperspektive impliziert vielmehr, daß die Gestaltung an **Entscheidungen** gekoppelt ist, denen sich nicht eindeutig bestimmte Programmcodes zuordnen lassen.

Die konsequente Durchsetzung dieser Gestaltungsperspektive in der Lernorganisation und in den Lernformen bietet die Chance, die Entwicklung einer adäquaten beruflichen Identität, wie wir sie mit der Chiffre des Prozeßmoderators beschrieben haben, als Nebenfolge des didaktischen Handelns zu unterstützen. Für die Lernorganisation als verzeitlichte sachlogische Struktur der Ausbildung heißt das:

1. Die Gestaltungsperspektive begleitet die Ausbildung kontinuierlich von ihrem Anfang bis zu ihrem Ende.

   Die Gestaltungsperspektive ist nicht die Kür einer Ausbildung, die erst nach der bestandenen Pflicht der Programmierfähigkeit eingenommen werden kann. Die Gestaltungsperspektive ist vielmehr unabhängig von der Kompetenz, auch nur eine Programmierzeile konstruieren zu können. Schon zu Beginn der Ausbildung können Softwareprodukte als gestaltete soziale Wirklichkeit betrachtet werden. Dies kann beispielsweise dadurch geschehen, daß am Computer Arbeitende danach befragt werden, welche Bedeutung sie dem Computer für ihre Arbeit beimessen.[130] Dies kann aber auch dadurch geschehen, daß eine alltägliche Sicht von Computerprogrammen mit Hilfe eines »fremden Blicks« destruiert wird: So kann z.B. die Auswirkung der Textprogramme auf die Art des Schreibens anhand von kritischen Texten unter einem ganz anderen Blickwinkel betrachtet werden.[131] Wie auch immer

---

[130] Solche Interviews haben z.B. die Auszubildenden (MTA) der Gesellschaft für Mathematik und Datenverarbeitung zu Beginn ihrer Ausbildung erfolgreich durchgeführt. Entscheidend für die Interviews ist ihre Konzentration auf die Arbeitsgestaltungsperspektive. Gefragt wird also z.B. nicht, was denn das Computerprogramm alles könne, bzw. wie es benutzt wird, sondern wie sich die Arbeit seit der Einführung des Programms verändert hat usw.

[131] Als ein solcher Text bietet sich z.B. [Hansen 1988] an.

die Gestaltungsperspektive konkret in die Ausbildung hineingetragen wird, entscheidend ist, daß sie von Anfang an zum kontinuierlichen Bezugspunkt der Ausbildung wird. Unser Qualifizierungsbaustein »Aufgabenverständnis« kann als Beispiel und Anregung zur Umsetzung der Gestaltungsperspektive in konkreten Unterrichtseinheiten herangezogen werden.

2. Eigenständige Unterrichtsfächer, die die berufliche Handlungskompetenz fördern, müssen von der Hegemonie des Programmcodes befreit werden.

   Z.B. wirft das Fach Mathematik immanent Fragestellungen auf (wie die Bestimmung der Formalisierbarkeit von Wirklichkeit, die, weil sie formalisierbar ist, noch lange nicht als formalisiert bestimmt werden kann), deren Reflexion dezidiert auf Möglichkeiten, Probleme und Grenzen der Gestaltung von Software verweist. Dagegen verstellt die Sichtweise, daß in der Mathematik nur rein syntaktisch-formale Strukturen entwickelt werden, deren Anwendung für ihr Wissen nicht konstitutiv ist, den Blick für die problematische Komplementarität, die die Mathematik auszeichnet: Die Anschauung und Anwendung kann nicht endgültig von mathematischen Begriffen getrennt werden. Die Aneignung von Mathematik erfolgt nicht über ontologisch fundierte Begriffe. Die prästabilierte Harmonie zwischen mathematischem Denken und realer Wirklichkeit, wie sie lange Zeit stillschweigend behauptet wurde, scheitert nicht nur an der natürlichen Wirklichkeit, die sich der Formalisierbarkeit durch mathematische Begriffe permanent widersetzt, sondern auch schon an der Erklärbarkeit des Verhaltens von Artefakten, wie z.B. Computern. Um zu verhindern, daß die Mathematik weiterhin als die »ontologische Letztbegründung« des Programmcodes mißverstanden wird, sollte das Verhältnis der Mathematik zur Informatik grundlegend reflektiert werden. Erst dann wird einsichtig, daß mathematisches Denken auch für die Gestaltungsperspektive von großer Bedeutung ist.

   Das Fach Software-Entwicklung ist von der Orientierung am Programmcode zu trennen und unter einer umfassenden Gestaltungsperspektive zu fassen. Da wir diese ausführlich mit unserem Rahmencurriculum »Moderne Software-Entwicklung« darstellen, verzichten wir hier auf eine nähere inhaltliche Beschreibung.

Zusammenfassend kann festgestellt werden: Das Aufbrechen der Hegemonie des Programmcodes durch **die Implementierung der Gestaltungsperspektive als Kontinuum der Ausbildungsorientierung** kann als Grundvoraussetzung dafür gelten, daß die Entwicklung einer adäquaten beruflichen Identität bei den Lernenden wahrscheinlicher wird. Die Implementierung der Gestaltungsperspektive bedeutet, daß **hervorgehobene Lernsituationen**, die die Gestaltung

als Resultat oder Notwendigkeit des Handelns erfahrbar machen, zu jedem Zeitpunkt der Ausbildung präsent sind, **Gestaltungserfahrungen gewissermaßen zu Interpunktionen der Ausbildung werden**. Neben die Umgestaltung der verzeitlichten Sachlogik der Ausbildung, das heißt letztendlich der praktischen Umsetzung eines neuen Leitbildes der Qualifizierung, tritt eine **Neubestimmung des funktionalen und inhaltlichen Verhältnisses zwischen einzelnen Unterrichtsfächern**. Als inhaltlichen Kern der Neubestimmung des Unterrichts der Software-Entwicklung schlagen wir das von uns entwickelte Rahmencurriculum »Moderne Software-Entwicklung« vor.

Ebenso wie die Lernorganisation nur dann zur Entwicklung einer angemessenen beruflichen Identität beiträgt, wenn sie sich von der Fixierung auf den Programmcode löst, können auch die unterschiedlichen Lernformen nur durch eine Umorientierung die Entwicklungschancen einer adäquaten beruflichen Identität befördern. Wir haben als Lernformen, die als unbeabsichtigte Nebenfolge das Erlernen einer bestimmten Identität begünstigen, die **Lösungsorientierung von Lernaufgaben, das Modellernen am Identitätsmodell des Lehrenden und die Arbeitstätigkeit im Praktikum** genannt. Ohne hier den Anspruch auf eine vollständige und systematische Beschreibung aller Lernformen aufstellen zu wollen, glauben wir doch zeigen zu können, daß die **Umorientierung dieser drei Lernformen** für die Unterstützung der Entwicklung einer angemessenen beruflichen Identität essentiell ist. Im einzelnen heißt dies:

1. Die Aufgabenstellungen sollen prinzipiell um Gestaltungsaufgaben gruppiert werden, solange keine gravierenden didaktischen Überlegungen dagegen sprechen. Dies gilt insbesondere für praktische Programmieraufgaben. Programmieren wird nicht mehr als Selbstzweck betrachtet, sondern immer nur als Mittel, um einen bestimmten Zweck, ein dem Programmieren äußerliches Gestaltungsziel zu erreichen. Falls bestimmte Programmierübungen (z.B. bestimmte grundlegende Algorithmen) nur zum Einüben einer bestimmten Programmierfähigkeit kontextlos dargeboten werden, so sind sie als solche kenntlich zu machen und im Verlauf der weiteren Ausbildung wieder mit der Gestaltungsperspektive »einzuholen«, das heißt zu verbinden. Grundsätzlich gilt, daß jedes Programmieren in irgendeiner Form mit der Gestaltungsfrage verbunden wird, wobei der Gestaltung das Primat zukommt.

   Insbesondere ermöglicht die Gestaltungsperspektive auch die kritische Aneignung vorgefundener Software. Während die Fixierung auf die Programmierung die Durchdringung komplexer Softwareprodukte oftmals als zum Scheitern verurteiltes Unternehmen erscheinen läßt, können mittels der

Gestaltungsperspektive schon zu Beginn der Ausbildung komplexe Softwareprodukte Gegenstände des Unterrichts werden.[132] Die Gestaltungsperspektive erschließt somit auch wichtige Aufgabenfelder des realen Arbeitsprozesses, die bisher mit dem Verweis auf ihre Komplexität (der Programmkonstruktion) gänzlich aus dem Unterricht ausgeschlossen waren. Die Gestaltungsperspektive ermöglicht in der Ausbildung einen adäquaten Umgang mit dem Komplexitätsproblem der Informatik.

2. Nur ein Lehrender, der nicht die Position eines »exemplarischen« Technikers einnimmt, kann als Identitätsmodell die Entwicklung einer angemessenen beruflichen Identität unterstützen. Kurz gesagt: Wenn ein Lehrer eine technikzentrierte Identität einnimmt, wird er, solange er sich nicht in Gegenreaktionen hervorrufende Widersprüche verstrickt, keine andere Identität vermitteln als eben eine technikzentrierte.

   Als adäquate Position des Lehrenden haben wir, angelehnt an Otte, die Position des exemplarischen Intellektuellen eingeführt. Auch wenn diese Position weder als Orientierungspunkt des Rollenhandelns noch als Identitätsmodell dem Lehrenden übergestülpt werden kann, so gibt es doch organisatorische Möglichkeiten, die einem latenten Vorhandensein dieser Position zum manifesten Durchbruch verhelfen. Eine solche organisatorische Möglichkeit stellt die Etablierung eines Kommunikationsrahmens und die Entwicklung einer »diskursfreudigen« Organisationskultur in der Ausbildungsinstitution dar.

3. Die Organisation des Praktikums muß konstruktiv das Dilemma lösen, daß die Praktikanten einerseits im ökonomischen Verwertungszusammenhang arbeiten, andererseits aber auch einen Einblick in den realen Arbeitsprozeß erhalten sollen. Die konstruktive Lösung dieses Dilemmas setzt eine Problemsicht voraus, die die Differenz zwischen den konkreten Arbeitstätigkeiten und den qualifikatorischen Anforderungen des realen Arbeitsprozesses thematisiert. Solche Differenzbegriffe sind z.B.:

   - Die **Komplexität des zu erstellenden Produkts**, die **Komplexität des sozialen Projektprozesses.** Der Praktikant bekommt oftmals die Erstel-

---

[132] Die Nonchalance, die IT-Fachkräfte gegenüber der vergegenständlichten Arbeit von Kollegen an den Tag legen, gilt als Eigenheit der Informatiker. Wenn wir allerdings einen Blick auf die IT-Ausbildung werfen, ist dies kaum verwunderlich: "Computer science education has prepared developers with a background that emphasises fresh creativity almost exclusively. Students learn to work alone and to develop programs from scratch. They are rarely asked to understand software systems they have not written" [Shaw z.n. Rout 1992, S. 31].

lung eines relativ eindeutig bestimmbaren Produkts übertragen. Die Komplexität des sozialen Projektprozesses ist schon jenseits seiner Arbeitstätigkeit durch die Projektteilnehmer reduziert worden.

Die Aufgabe des Praktikanten sollte daher darin bestehen, die ihm zugeteilte Arbeitstätigkeit qualifiziert auszuführen und die **Komplexität des Projektes zu rekonstruieren**. Als heuristische Mittel kann er dabei auf verschiedene weitere Begriffe zurückgreifen: die **Projektbeteiligten**, die **Projektbetroffenen**, das **Projektziel** und das Verhältnis zu den Projektbeteiligten, die **Arbeitsteilung** im Projekt, die **eigene Arbeitstätigkeit** und ihre Bedeutung für das Projekt, die **Projektorganisation** (**Projektmodell**, **Kooperationsformen**), der **Projektumfang** und ähnliche Begriffe (vgl. auch Rahmencurriculum »Moderne Software-Entwicklung«).

- **Reflektierende Selbstbeobachtung der eigenen Arbeitstätigkeit.** Der Praktikant sollte zur Selbstreflexion seiner eigenen Tätigkeit angehalten werden. Auch dazu sollten bestimmte, in ihrer Zahl limitierte heuristische Begriffe zur Verfügung stehen, um eine Diffusion der Aspekte der Selbstbeobachtung zu vermeiden. Als solche schlagen wir vor: die funktionale Bestimmung der eigenen Arbeitstätigkeiten bezüglich der eigenen Arbeitsaufgabe (1); die funktionale Bestimmung der eigenen Aufgabe im Kontext des Gesamtprojektes bzw. des Arbeitsprozesses der Benutzer (2); die Bewertung der eigenen Qualifikation (3); die Beschreibung tatsächlicher und potentieller (aus pragmatischen Gründen nicht ausgetragener) Konflikte (4).

  Da man nicht gleichzeitig handeln und sich selbst beobachten kann, ist es u.E. sinnvoll, die oben genannten Aspekte zu konkreten Fragestellungen zu operationalisieren, die der Praktikant leichter handhaben kann. Zu (1) kann z.B. angeregt werden, daß der Praktikant seine Arbeitstätigkeiten in eine Verkettung von »Um-Zu-Aussagen« auf seine Arbeitsaufgabe bezieht. Ähnlich kann auch (2) behandelt werden, wobei hier auf die Struktur des »praktischen Syllogismus« zurückgegriffen werden kann.[133] Die Bewertung der eigenen Qualifikationen (3) kann

[133] Der praktische Syllogismus zerlegt die Interpretation von Handlungen in drei Schritte: 1. Wollensaussage (der Handelnde beabsichtigt den Zustand y herbeizuführen); 2. Wissensaussage (der Handelnde ist überzeugt, y nur herbeiführen zu können, wenn er x ausführt); 3. Handlungsaussage (der Handelnde führt x aus). Insbesondere sollte der Praktikant die Wollens- und Wissensaussagen im Projekt durch Nachfragen eruieren.

skaliert strukturiert werden. Welche Ausbildungsinhalte habe ich am meisten brauchen können, welche am wenigsten? Oder: Wo fühlte ich mich am sichersten, wo fühlte ich mich am unsichersten (überfordert)? Zu (4) können auch konkrete Fragen konstruiert werden: Was habe ich gerne gemacht, was ungerne? Was hätte ich anders gemacht, wenn ich keine Vorgaben bekommen hätte? Welche Konflikte habe ich gehabt? Welche Konflikte habe ich bemerkt und wie wurde mit diesen umgegangen?

Die eben genannten Differenzbegriffe sollten zu einer strukturierten Vor- und Nachbereitung des Praktikums verwendet werden. Sie ermöglichen es u.E., das Dilemma des Praktikums, gleichermaßen Arbeitsrealität zu sein und - im Sinne einer beschränkten Verallgemeinerungsfähigkeit - nicht zu sein, didaktisch konstruktiv zu bearbeiten. Eine direkte Nebenfolge dieser didaktischen Vor- und Nachbereitung des Praktikums ist die Unterstützung der Entwicklung einer angemessenen beruflichen Identität. Auch hier wird gewissermaßen eine **Gestaltungsperspektive in das Praktikum implementiert**, das, bedingt durch seine beschränkte, abgekapselte Arbeitsaufgabe, von sich aus oftmals nur einen eingeengten Gestaltungshorizont bietet.

Zusammenfassend können wir sagen: Ebenso wie die verzeitlichte Lernorganisation durch eine hegemoniale Gestaltungsperspektive, die an Stelle des Programmcodes treten soll, umzugestalten ist, sollte auch in den Lernformen die Gestaltungsperspektive implementiert werden.

Erst wenn die Gestaltungsperspektive die Ausbildung hegemoniert, kann die Entwicklung einer angemessenen beruflichen Identität als Nebenfolge des didaktischen Handelns erwartet werden.

6.3 Didaktisches Handeln, das direkt zur Entwicklung einer unangemessenen Identität beiträgt

Wir haben das berufliche Identitätslernen als Herstellung einer Balance zwischen Wissen und Fähigkeiten, angenommenem Berufsbild (Berufsrolle und berufliche Aufgaben) und dem Selbstbild des Lernenden gekennzeichnet. Entsprechend wollen wir didaktisches Handeln, das direkt die Förderung der beruflichen Identität zum Ziel hat, dadurch charakterisieren, daß es sich aktiv auf das Verhältnis zwischen den auszubalancierenden Faktoren bezieht. Als didaktische Zielgrößen, die in diesem Sinne direkt auf die Entwicklung der beruflichen Identität Einfluß nehmen, können folgende genannt werden:

- Motivationen

- Einstellungen
- Fundamentale Ideen des Aufgabenverständnisses

Motivationen

Die didaktische Herstellung von Motivation impliziert prinzipiell, daß sich der Lernende mit der Aufgabe identifiziert, also intrinsische Motivationen mobilisiert. »Der Lehrende packt den Lernenden bei seinem Ehrgeiz.« Wenn wir diese alltagssprachliche Ausdrucksweise näher betrachten, erhalten wir so ziemlich genau das, was motivierendes didaktisches Handeln bedeutet:

1. Der Lernende hat einen Ehrgeiz, der sich vorerst allerdings nicht auf die zu lösende Aufgabe bezieht. Also muß man ihn erst bei seinem Ehrgeiz packen.
2. Dies geschieht dadurch, daß der Lehrende die Aufgabe auf das angenommene Objekt des Ehrgeizes des Lernenden bezieht.

Mit Hilfe dieser »kleinen« Verschiebung versucht er, die Aufgabenstellung dem Selbstbild der Lernenden näherzubringen, ohne dieses selbst in Frage zu stellen. Z.B.: Die Lernenden sitzen lustlos vor einer Lernaufgabe der Anforderungsanalyse, als deren Lösungsweg die strukturierte Zerlegung vom Lehrenden vorgesehen wurde. Der Lehrer weiß, daß die Lernenden großen Ehrgeiz entfalten, wenn Aufgabenstellungen durch strukturiertes technisch-logisches Denken zu lösen sind. Er behauptet nun, daß die von ihm gestellte Aufgabe nur durch logisches Denken gelöst werden kann. Falls ihm dies gelingt, und die Aufgabe tatsächlich allein durch die Logik der strukturierten Analyse lösbar ist (zumindest **eine** Lösung kommt zustande), stellt er aktiv ein bestimmtes Verhältnis zwischen der Aufgabenstellung der Anforderungsanalyse und dem Selbstbild der Lernenden her (genauso kann ein Lehrer schon die Aufgabenstellungen nach den angenommenen Motivationslagen der Lernenden entwerfen: »Ich baue in meine Aufgaben immer so kleine Tüftelprobleme ein, das hält die Schüler bei der Stange und gibt ihnen Erfolgserlebnisse«).

Damit haben wir auch schon eine Form des didaktischen Handelns, das **negative Formen** der Identitätsentwicklung aktiv unterstützt, beschrieben: **Vorhandene Motivationslagen der Lernenden werden didaktisch mobilisiert**, auch wenn die mit den Motivationen verbundenen Sichtweisen der Aufgabe nur unzureichend gerecht werden.

Einstellungen

Der Begriff der Einstellungen ist ein in der Psychologie und Soziologie nicht eindeutig bestimmter Begriff. Wenn wir diesen Begriff hier verwenden, so legen wir auf zwei Aspekte, die mit diesem Begriff verbunden werden, besonderen Wert:

- "Einstellung ist die Art (Akzentuierung, Tönung) der Gerichtetheit, der Ausrichtung, des selektiven Vorgehens (z.B. bei der Lösung einer Aufgabe)"[Dorsch 1982, S. 163]. Als solche »Art« hat die Einstellung eine **determinierende Tendenz**.
- "Einstellung ist die seelische Haltung gegenüber einer Person, einer Idee oder Sache, verbunden mit einer Wertung oder einer Erwartung" [ebd., S. 163]. Als seelische Haltung kann der Begriff der Einstellungen als **psychisches Korrelat sozialer Wertorientierungen und Rollenerwartungen** bezeichnet werden.

Beide Formen der Einstellungen werden im didaktischen Handeln zum größten Teil **positional** vermittelt, das heißt, sie werden nicht Gegenstand eines »herrschaftsfreien« Aushandlungsprozesses, sondern sie werden über die **strukturelle Überlegenheit des Lehrenden** transportiert. In folgendem wollen wir nur dasjenige didaktische Handeln betrachten, in dem Einstellungen direkt zum Gegenstand des didaktischen Handelns werden und nicht nur als Nebenfolgen desselben anzusehen sind. Dazu werden wir zwei Beispiele kurz skizzieren, die jeweils einen Aspekt des Einstellungsbegriffs thematisieren.

Einstellungen als Lernhilfen - die determinierende Tendenz von Einstellungen:

Oftmals erscheint es dem Lehrenden sinnvoll zu sein, auf Problemlöseschwierigkeiten der Lernenden mit Einstellungsempfehlungen zu reagieren. **Einstellungen werden damit als Lernhilfen angeboten, die es dem Lernenden ermöglichen sollen, durch die Übernahme einer neuen Problemsicht aktuelle Lernhemmungen selbständig zu überwinden.** Z.B.: Der Lehrende bemerkt, daß die Lernenden mit ihrer Einstellung, das Problem als »technisches Tüftelproblem« selektiv wahrzunehmen, nicht weiterkommen. Er insistiert daher auf eine Einstellung, die wir als **»technisch distanzierte Rationalität«** bezeichnen können. Er lenkt die Aufmerksamkeit auf Struktureigenschaften des Problems, die technisch rational klassifizierbar sind und so auf Umwegen eine systematische Lösung des Problems eröffnen. Wenn die technische distanzierte Rationalität allerdings nicht zur Lösung des Problems führt, empfiehlt er die Einstellung des **»geduldigen technischen Tüftelns«,** gepaart mit einem Appell an die Kreativität der Lernenden. Damit leistet der Lehrende direkt einen Beitrag zur Entwicklung einer technikzentrierten Identität, die selbst dann noch aufrecht er-

halten werden kann, wenn das Problem sich der technischen Logik entzieht. Das Oszilieren der Einstellungen zwischen »technisch distanzierter Rationalität« und »geduldigem technischen Tüfteln« begünstigt eine berufliche Identität des »Technikers«, die in bestimmten Situationen zur Identität des »Hackers« regrediert [vgl. Johnson 1988, S. 147ff.].

Einstellungen als Vermittlung grundsätzlicher Wertorientierungen:

Wertungen und Erwartungen, die sich auf Personen, Ideen und Sachen beziehen, sind Grundmuster didaktischen Handelns. Wie bestimmte Wertorientierungen die Sichtweise der gesamten Fachdidaktik maßgeblich prägen, kann exemplarisch an der Diskussion um den Informatikunterricht für die Sekundarstufe II verdeutlicht werden.[134] Dazu wollen wir zwei gegensätzliche Positionen, die die aktuelle Diskussion bestimmen, gegenüberstellen[135]:

"Als ordnendes Prinzip wird der Bezug des Menschen zum Computer grundgelegt. Es ermöglicht die systematische Behandlung von Informatik-Grundlagenwissen zusammen mit Fragen der Sozialverträglichkeit des Werkzeugs Computer, rückt die Computertechnik in einen Zusammenhang mit ihrem Anwender und Gestalter und betont die Wechselwirkung zwischen der Technik und dem Menschen." (Empfehlung des Arbeitskreises »Informatik in der Sekundarstufe II« der GI).

Gegen diese Position hat sich dagegen vehement der Informatiklehrer und Fachdidaktiker Baumann ausgesprochen. Er sieht darin eine »Kolonisierung der Informatik durch Soziologie und/oder Politologie«, der entschieden entgegen getreten werden müsse. Es gehe sogar darum, die Informatik wieder von dem Einfluß der Gesellschaftswissenschaften zu befreien. Einen neuen Aufsatz beschließt Baumann mit der Forderung, die Informatik "nicht in eine technikorientierte Sozialkunde um(zu)funktionieren oder gar zur Vermittlung politischer Bekenntnisse mißbrauchen (zu) lassen."

Daß diese unterschiedlichen Einstellungen zu grundsätzlich unterschiedlichen Wertorientierungen und Erwartungen im praktischen didaktischen Handeln führen, ist leicht nachzuvollziehen.

---

[134] Diese Diskussion hat für uns nicht nur exemplarischen Wert, da im Unterricht der Sekundarstufe II zunehmend die grundlegende Sichtweise der Informatik vermittelt wird, auf die sich die nachfolgende Ausbildung beziehen muß. Nicht selten sind es gerade diese schulischen Vorerfahrungen mit der Informatik, die zur Berufswahl der Lernenden geführt haben.

[135] Die Positionen werden zitiert nach [Hauf-Tulodziecki 1993].

Fundamentale Ideen des Aufgabenverständnisses

Durch die Implementierung fundamentaler Ideen des Aufgabenverständnisses in die Ausbildung werden Kristallisationspunkte geschaffen, auf die sich die Identitätsentwicklung der Lernenden direkt ausrichten kann. Wenn wir nach fundamentalen Ideen in der Ausbildung suchen, die die Entwicklung einer unangemessenen Identität unterstützen, werden wir kaum fündig. Dies liegt daran, daß die IT-Ausbildung gerade von einem **Mangel an fundamentalen Ideen** geprägt ist. Jeweils sich den neuesten technischen Anforderungen anpassend, gleicht die Ausbildung (und insbesondere die Weiterbildung) einem **unstrukturierten Flickenteppich**. Statt bewußt ein Leitbild der Ausbildung in fundamentale Ideen didaktisch umzusetzen, findet sich oftmals nur eine »naturwüchsige« Orientierung am Programmiercode. Die didaktische Umsetzung fundamentaler Ideen des Aufgabenverständnisses werden wir daher erst im folgenden diskutieren, wenn wir unsere didaktischen Vorschläge zur Unterstützung der Entwicklung einer angemessenen beruflichen Identität vorstellen werden.

Zusammenfassend kann festgestellt werden: Identitätsentwicklung wird meist dann direkt angestoßen, wenn Motivationen und Einstellungen zum Bezugspunkt des didaktischen Handelns werden. Motivationen und Einstellungen sind miteinander und mit der (vor-)beruflichen Identität der Lernenden auf das engste verbunden. Fundamentale Ideen, die die Entwicklung einer angemessenen beruflichen Identität fördern würden, bilden kaum Kristallisationspunkte einer IT-Ausbildung, die sich nach dem Prinzip des Flickenteppichs weiterentwickelt. Didaktisches Handeln, das sich die Entwicklung einer angemessenen beruflichen Identität zum Ziel setzt, muß andere Einstellungen und Motivationen didaktisch unterstützen und fundamentale Ideen des Aufgabenverständnisses didaktisch umsetzen.

6.4 Didaktisches Handeln, das die Entwicklung einer angemessenen beruflichen Identität unterstützt - fundamentale Ideen im didaktischen Kontext

Während wir die Dimensionen der Motivation und der Einstellungen hauptsächlich eingeführt haben, um die **didaktische Aufmerksamkeit** auf Handlungsformen zu lenken, die direkt die Entwicklungsrichtung des Identitätslernen beeinflussen, wenden wir uns nun einem **programmatischen Konzept** zu, welches für sich beansprucht, eine angemessene Identitätsentwicklung aktiv und effektiv zu unterstützen. Die Reflexion des didaktischen Umgangs mit Motivationen und Einstellungen ist unerläßlich, um die **Kohärenz im didaktischen Handeln** herzustellen, aber sie liefert kein didaktisch **schlüssiges Konzept des Identitätslernens**. Ein solches Konzept stellt dagegen der **Ansatz der fundamentalen**

Ideen[136] dar, der im praktischen didaktischen Handeln wiederum nur kohärent umgesetzt werden kann, wenn er den Problemen der Motivation und der Einstellungen Beachtung schenkt.

Dem didaktischen Konzept der fundamentalen Ideen kommt nicht nur eine besondere Rolle bei der Konzeption einer neuausgerichteten IT-Ausbildung zu, sondern es unterstützt auch in besonderem Maße den **Prozeß der Neuausrichtung** der bisherigen IT-Ausbildung. Dieser Umstellungsprozeß läuft prinzipiell Gefahr, daran zu scheitern, daß er entweder zu radikal konzipiert wird und daher in den bestehenden Ausbildungsorganisationen mit dem vorhandenen Lehrkörper praktisch nicht umgesetzt werden kann, oder daß er nur halbherzig angegangen und in Randbereiche der Ausbildung gedrängt wird. Als Resultat von letzterem wird das Ziel einer Neuausrichtung der Ausbildung vollständig verfehlt: Das neue Qualifizierungsziel wird zu einer bezugslosen Und-Qualifikation, deren Relevanz für die berufliche Handlungskompetenz gegen Null strebt. Die Auszubildenden entwickeln keine den Aufgaben angemessene berufliche Identität, sondern sie werden bestenfalls zu Technikern mit schlechtem Gewissen, die gelernt haben, daß sie eigentlich auch die Arbeitsgestaltung berücksichtigen müßten, aber nie erfahren haben, in welchem Verhältnis dieser »kategorische Imperativ« zu ihrer praktischen Arbeitstätigkeit steht. Um dieses zu verhindern, müssen grundlegende Bestimmungen des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« als fundamentale Ideen in den bestehenden Unterricht integriert werden. Die Funktion dieser fundamentalen Ideen besteht nicht nur darin, an hervorgehobenen Positionen auf Probleme der Arbeitsgestaltung hinzuweisen, sondern auch in der Verfolgung des Ziels, bestehende Unterrichtseinheiten auf einen neuen Fokus, die Gestaltung von Arbeit, auszurichten.

Als fundamentale Ideen begreifen wir zum einen die Idee, daß Software-Entwicklung primär **Arbeitsgestaltung** ist, und zum anderen die Idee, daß Software-Entwicklung nur als **kooperative Gestaltungsaufgabe** angemessen bewältigt werden kann. Beide Ideen scheinen auf den ersten Blick sehr einfach zu sein. Wenn wir allerdings bedenken, daß beide Ideen auf die Entwicklung einer neuen beruflichen Identität einer Berufsgruppe abzielen, sehen wir die didaktischen Anstrengungen, die eine Umsetzung dieser scheinbar so einfachen Ideen erfordert.

---

[136] Der Ansatz der »fundamentalen Ideen« geht auf Bruner zurück und wurde von Schwill zur Strukturierung des Informatikunterrichts der Sekundarstufe II aufgegriffen [vgl. Bruner 1970; Schwill 1993; Hauf-Tulodziecki 1994].

Die didaktische Umsetzung fundamentaler Ideen

Die didaktisch-methodische Umsetzung dieser fundamentalen Ideen in der bestehenden Ausbildungspraxis verlangt die Planung und Durchführung ausstrahlungsfähiger Lernerfahrungen.

Ausstrahlungsfähige Lernerfahrungen wirken als Interpunktionen des Ausbildungsgangs. Sie sind Ausgangs- und Bezugspunkt von Unterrichtseinheiten. Das Verhältnis von ausstrahlungsfähigen Lernerfahrungen und nachfolgenden Unterrichtseinheiten ist spiralförmig organisiert. Diese Spiralförmigkeit beruht darauf, daß die nachfolgenden Unterrichtseinheiten in einfacher Form schon in den ausstrahlungsfähigen Lernerfahrungen vorweggenommen werden. Diese Vorwegnahme reicht von der Erfahrung eines spezifischen Problemhorizonts bis hin zu exemplarischen Problemlösungen.

Das Verhältnis von einer ausstrahlungsfähigen Lernerfahrung zur nächsten muß durch Anschlußfähigkeit gekennzeichnet sein. Dabei gibt es prinzipiell zwei Möglichkeiten:

- Die nachfolgende Lernerfahrung kann die vorhergehende enthalten, indem sie ihr gegenüber eine Erweiterung oder Vertiefung darstellt.
- Die nachfolgende Lernerfahrung kann eine andere Perspektive einführen oder betonen.

Damit Lernerfahrungen allerdings wirklich zu integrierbaren, subjektiven Lernerfahrungen werden können, müssen sie einen Bezug zur Lebenswelt der Lernenden enthalten. Gerade ausstrahlungsfähige Lernerfahrungen sind didaktisch konstruierte Lernsituationen, die eine spezifische Problemperspektive des Arbeitsprozesses evident machen, indem sie diese mit lebensweltlichen Erfahrungen der Lernenden verbinden. Die Konstruktion dieser Lernsituationen setzt eine theoretische Analyse des Arbeitsprozesses ebenso voraus wie die Antizipation der Lernfähigkeiten der Lernenden.

Zur Verdeutlichung der praktischen Umsetzung des Konzepts der fundamentalen Ideen zur Neuausrichtung der IT-Ausbildung werden wir abschließend drei Beispiele kurz skizzieren. Die ersten beiden Beispiele wurden im MTA-Ausbildungsgang der GMD im ersten Ausbildungshalbjahr durchgeführt. Das dritte Beispiel kann als eine mögliche, diesen Beispielen folgende ausstrahlungsfähige Lernerfahrung angesehen werden.

Die fundamentale Idee der Arbeitsgestaltung als ausstrahlungsfähige Lernerfahrung - Beispiel: Interviews

Ziel dieser Lernerfahrung war es, die Lernenden schon zu Beginn der Ausbildung mit der fundamentalen Idee, daß Software-Entwicklung Arbeitsgestaltung ist, vertraut zu machen.

Für eine überdauernde Lernerfahrung erscheint in dieser Phase der Ausbildung eine Annäherung über problematisierende Texte zu abstrakt. Vielmehr ging es darum, das Phänomen der Arbeitsgestaltung an der sinnlichen Wahrnehmung lebendiger Arbeit lebensweltlich nachvollziehbar zu machen. Die Durchführung einer Software-Entwicklungsaufgabe, die dezidiert das Phänomen der Arbeitsgestaltung in den Mittelpunkt der Aufgabenstellung rücken würde, war zu Beginn der Ausbildung allein aufgrund fehlender Programmierkenntnisse unmöglich.

Aus diesen Gründen wurde eine Lernaufgabe konzipiert, die auf strukturierten Interviews von Softwarebenutzern und Entwicklern beruht. Die Fragen an die Benutzer zielten auf die Brauchbarkeit der Software und auf die Veränderungen ihrer Arbeit, verursacht durch die Einführung der Software.

Neben den Interviews mit den Benutzern wurden auch Befragungen von Software-Entwicklern durchgeführt, die das Verhältnis von Entwicklern und Benutzern im Software-Entwicklungsprozeß zum Gegenstand hatten. Damit gerieten nicht nur die Software-Entwickler als Gestalter von Arbeit in das Blickfeld, sondern auch Ansätze der fundamentalen Idee der Kooperation. Gleichzeitig bildete die Interviewsituation selbst schon eine erste, nicht immer unproblematische, direkte Kooperationserfahrung.

Die Resonanz, die diese Lernerfahrung bei den Lernenden erfuhr, zeigte sich nicht nur in einer motivierten Nachbereitung, bei denen die einzelnen Interviewgruppen ihre Ergebnisse vorstellten, sondern auch darin, daß noch lange nach der Durchführung der Interviews auf diese Lernerfahrungen Bezug genommen wurde.

Die fundamentale Idee der Kooperation als ausstrahlungsfähige Lernaufgabe - Beispiel: Arbeitsteilige Bearbeitung einer mathematischen Programmieraufgabe

Dieses Beispiel folgte der oben beschriebenen Lernerfahrung. Auf den ersten Blick scheint die arbeitsteilige Lösung einer einfachen mathematischen Programmieraufgabe (Polynomaufgabe) ein Rückschritt zu sein, der eher dem Fokus des Programmcodes als dem Fokus der Arbeitsgestaltung dient. Eine nähere Betrachtung des Beispiels zeigt allerdings, daß die spezifische didaktische Organisation der Lernaufgabe gerade den Fokus auf den Programmcode verhinderte.

Die Lernaufgabe bestand in der arbeitsteiligen Lösung einer einfachen mathematischen Programmieraufgabe. Die Struktur der Arbeitsteilung war von vornherein so festgelegt, daß einzelne Lerngruppen eine deutlich unterscheidbare Perspektive bezüglich der Aufgabe einnehmen mußten. Es gab eine Testgruppe, eine Dokumentationsgruppe, eine Koordinationsgruppe und eine Benutzerschnittstellengruppe. Darüber hinaus wurde die Programmieraufgabe arbeitsteilig bearbeitet. Die sich bei der Bewältigung dieser Aufgabe ergebenden Kooperationsprobleme mußten zwangsläufig in einem intersubjektiven Kommunikationsprozeß gelöst werden. Die Entwickler-Entwickler-Kooperation ähnelte bei dieser Lernaufgabe strukturell der Anwender-Entwickler-Kooperation. Die dabei auftretenden Konflikte konnten nicht durch formale Regeln verdeckt werden, da sie den Lernenden zum Zeitpunkt des Lernprojektes nicht zur Verfügung standen. Die Lernerfahrungen, die bei der Bewältigung dieser Lernaufgabe gemacht wurden, konnten in den nachfolgenden Unterrichtseinheiten wieder aufgenommen und vertieft werden.

Die Integration der fundamentalen Ideen der Arbeitsgestaltung und der Kooperation - Beispiel: Praktische Software-Entwicklungsaufgabe

Konzeptionell könnte dieser ausstrahlungsfähigen Lernerfahrung einer arbeitsteiligen Programmieraufgabe eine Lernerfahrung folgen, die eine enge, lebensweltlich nachvollziehbare Verschränkung der fundamentalen Idee der Kooperation und der Arbeitsgestaltung bewirkt.

Eine solche Lernerfahrung könnte beispielsweise in der Entwicklung einer eigenen Arbeitsumgebung am Computer für die Lernenden bestehen. Die didaktische Organisation einer derartigen Software-Entwicklungsaufgabe bestünde ebenso in der strikten Zuweisung von Entwickler- und Benutzerrollen, wie in der Einforderung exakter Dokumentationen, die sich auch auf den Kooperationsprozeß bezögen.

Das didaktische Konzept der fundamentalen Ideen in der Weiterbildung

Das didaktische Konzept der fundamentalen Ideen beschränkt sich keineswegs nur auf die Gestaltung der Ausbildung, sondern ist auch für den Bereich der betrieblichen und außerbetrieblichen Weiterbildung in modifizierbarer Form verwendbar. Während das Konzept der fundamentalen Ideen in der Ausbildung dahingehend unterrichtspraktisch realisiert wird, daß ausstrahlungsfähige und handlungsorientierte Lernerfahrungen konzipiert werden, die diesen Ideen Rechnung tragen, so verhält es sich in der Weiterbildung genau umgekehrt: Problematische Projekterfahrungen der Teilnehmer aus der praktischen Arbeit

werden zum Ausgangspunkt einer gemeinsamen Ursachenanalyse. Über die Analyse kritischer Handlungssituationen[137], deren Bewältigung oftmals über den Erfolg oder das Scheitern ganzer Projekte entscheiden, werden die fundamentalen Ideen der Arbeitsgestaltung und der Kooperation aus praktischen Arbeitserfahrungen rekonstruiert. Erst diese Herleitung fundamentaler Ideen der Software-Entwicklung aus den praktischen Arbeitserfahrungen ermöglicht eine systematische Qualifizierung in den Feldern Arbeitsgestaltung und Kooperation, eine Qualifizierung, die nicht zu oberflächlichem Wissen führt, sondern mit einer grundsätzlichen Umorientierung der beruflichen Identität einhergeht.

Mit dieser kurzen Skizze des didaktischen Konzepts der fundamentalen Ideen schließen wir unsere Ausführungen zum didaktischen Handlungsbereich der Entwicklung einer angemessenen beruflichen Identität.

6.5 Zusammenfassung: Didaktisches Handeln und die Entwicklung der beruflichen Identität

Ausgangspunkt unserer Überlegungen zur didaktischen Unterstützung der Entwicklung einer angemessenen beruflichen Identität war das Phänomen, daß sich berufliche Identitäten auf zwei unterschiedlichen didaktischen Ebenen bilden. Einerseits ist Identitätslernen eine permanente Nebenfolge von sachbezogenen Lernprozessen. Wir können hier auch vom indirekten Identitätslernen sprechen. Andererseits werden im praktischen Unterricht auch direkt Einstellungen und Motivationen vermittelt, die Ausdruck einer bestimmten Auffassung von beruflicher Identität sind.

Beide Formen der Identitätsvermittlung werden in der gängigen Unterrichtspraxis in der Regel nicht reflektiert oder bewußt zur Unterstützung einer bestimmten Identitätsentwicklung didaktisch eingesetzt. Die Nichtzuständigkeit der Didaktik für Fragen der beruflichen Identitätsbildung bleibt unproblematisch, solange die Lernorganisation und die Lernformen mit der sachlogischen Orientierung der Ausbildung konform geht. Die Unterstützung der Identität des Technikers muß kaum bewußt in Angriff genommen werden, da sie strukturell tief in der Ausbildungstradition festgeschrieben ist und auch dem allgemeinen Bild des Informatikers in der Gesellschaft entspricht. Die Auswüchse des Hackertums bildeten die einzigen »Identitätskonflikte«, denen sich die traditionelle IT-Ausbildung bisher stellen mußte.

---

[137] Diese kritischen Handlungssituationen finden sich ausführlich im Kap.IV unseres Rahmencurriculums »Moderne Software-Entwicklung« wieder.

Um das Qualifizierungsziel »Ganzheitliche Arbeitsgestaltungskompetenz« didaktisch umsetzen zu können, ist es notwendig, die klassische, tief in der Ausbildungsstruktur verwurzelte, technikzentrierte Identität zu überwinden. Als Ausgangspunkt dieser Überwindung haben wir die Formen negativer Identitätsstiftung in der vorhandenen Ausbildung analysiert, um ihnen mit bewußtem didaktischem Handeln entgegentreten zu können. Diese Formen reichen von der hegemonialen Ausrichtung der Ausbildung am Programmcode bis hin zur Vermittlung bestimmter Einstellungen, Wertorientierungen und Motivationen in konkreten didaktischen Handlungssituationen.

Neben dem Ausgangspunkt der didaktischen Unterstützung einer angemessenen beruflichen Identität, der in der Vermeidung unbewußter technikzentrierter Identitätsstiftung besteht, haben wir einige Ansatzpunkte einer angemessen Unterstützung der beruflichen Identitätsentwicklung dargestellt. Diese reichen von der Implementierung der Gestaltungsperspektive als Kontinuum der Ausbildungsorientierung über die Reflexion des beruflichen Aufgabenverständnisses bis hin zur Gestaltung ausstrahlungsfähiger Lernerfahrungen, die den fundamentalen Ideen Arbeitsgestaltung und Kooperation gerecht werden.

Der didaktische Handlungsbereich des Identitätslernens ist bisher in der berufspädagogischen Diskussion weitgehend vernachlässigt worden. Einerseits, weil er oftmals für die Entwicklung der beruflichen Handlungskompetenz als marginal angesehen wurde, andererseits, weil Identitätsentwicklung - und sei es auch nur die berufliche - als etwas Privates gilt, das von der direkten didaktischen Beeinflussung freigehalten werden sollte. Wir hoffen gezeigt zu haben, daß die bewußte didaktische Unterstützung einer angemessenen beruflichen Identität nicht nur für die Entwicklung der beruflichen Handlungskompetenz von zentraler Bedeutung ist, sondern auch den Individuen Entfaltungsmöglichkeiten bietet, die über die Enge einer technikzentrierten beruflichen Identität hinausgehen.

7 Resümee: Subjektorientierte Didaktik und Ganzheitliche Arbeitsgestaltungskompetenz

Abschließend wollen wir kurz den Argumentationsgang rekapitulieren, der unsere didaktischen Überlegungen zur Vermittlung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« geleitet hat.

Die Fundierung einer Didaktik der Ganzheitlichen Arbeitsgestaltungskompetenz stellt uns vor Probleme, die auf den ausgetretenen Pfaden herkömmlicher Fachdidaktiken nicht zu lösen sind. Der Versuch, eine Fachdidaktik Ganzheitliche Arbeitsgestaltungskompetenz als hierarchisch gegliederte Wissensmenge zu konzipieren, die den Lernenden möglichst effizient vermittelt werden soll, muß an

den Ansprüchen des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« scheitern. Eine didaktische Abbildung der mannigfaltigen beruflichen Handlungssituationen ist nicht möglich. Vielmehr gilt es, grundlegende subjektive Regulative der beruflichen Handlungskompetenz zu vermitteln. Als grundlegende Regulative haben wir im ersten Teil unserer Argumentation das Wissen und den subjektiven Umgang mit Wissen, soziale Kompetenzen und die Ausprägung einer spezifischen beruflichen Identität analysiert. Diese Analyse der beruflichen Handlungskompetenz findet ihren Niederschlag in der Konzeption einer subjektorientierten Didaktik, die den zweiten Teil unserer didaktischen Argumentation bildete. Wir hoffen gezeigt zu haben, daß sich eine subjektorientierte Didaktik nicht auf unverbindliche modische Postulate wie das »Lernen des Lernens« beziehen muß, sondern sich dezidiert auf die beruflichen Handlungssituationen von IT-Fachkräften stützen kann. Mit zahlreichen praktischen Vorschlägen zum didaktischen Handeln im Unterricht, aber auch zur Unterrichtsplanung, haben wir konkrete Anregungen zur Veränderung des bestehenden Unterrichts formuliert.

Trotz der zahlreichen Informatikfachbezüge kann eine subjektorientierte Didaktik kaum den Anspruch erheben, eine umfassende Fachdidaktik zu ersetzen. Vielmehr bildet die subjektorientierte Didaktik die notwendige methodische Fundierung der Umsetzung der Fachdidaktik.

Die Umsetzung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« steht gewissermaßen auf zwei Beinen: dem didaktisch-methodischen Konzept einer subjektorientierten Didaktik und dem curricularen Entwurf von fundamentalen Ideen Ganzheitlicher Arbeitsgestaltungskompetenz. Dabei handelt es sich keineswegs um zwei monolithische Blöcke, die von den Lehrenden erst mühsam aufeinander bezogen werden müssen. Vielmehr weisen beide Blöcke zahlreiche systematische Beziehungen zueinander auf. Diese Beziehungen fußen auf dem gemeinsamen Ausgangspunkt: der Analyse des Arbeitsprozesses der Software-Entwicklung. Die Komplementarität unserer bisherigen Überlegungen mit der curricularen Darstellung der Grundzüge einer Fachdidaktik der Ganzheitlichen Arbeitsgestaltungskompetenz ist daher keineswegs zufällig.

Die curriculare Beschreibung des Qualifizierungsziels »Ganzheitliche Arbeitsgestaltungskompetenz« bildet das Thema der folgenden Kapitel.

Teil B
Rahmencurriculum »Moderne Software-Entwicklung«

Einleitung

Software Engineering als »Scharnierfach«

Das Themengebiet »Software Engineering« wird seit einigen Jahren als Unterrichtsbestandteil der Ausbildung von IT-Fachkräften angeboten. Dort, wo die Ausbildung nach einem Fächerkanon organisiert ist, findet sich »Software Engineering« oft sogar als eigenständiges Fach. Eine Betrachtung der Historie des Fachgebiets, so wie wir sie anhand unserer Expertengespräche mit Verantwortlichen und Dozenten aus Ausbildungsinstitutionen nachgezeichnet haben, macht deutlich: Das Fach ist entstanden, weil die in der Ausbildung vermittelten Lehrinhalte (Sprachen, Grundlagen etc.) gegenüber der beruflichen Praxis von IT-Fachkräften nicht mehr ausreichend anschlußfähig waren. Von Anfang an war das Fach daher darauf angelegt, ein **Verbindungsglied** zwischen den in der Ausbildung vermittelten Inhalten und der betrieblichen Praxis zu sein.

Der Unterrichtsgegenstand »Software Engineering« wird in Ausbildungs- und Umschulungsgängen vor allem in Reaktion auf eine verkürzte Sichtweise der Schüler bezüglich der Programmierung angeboten. Diese ist in der Logik der Vermittlung von Programmierkenntnissen geradezu zwingend angelegt: Die Schüler lernen in den ersten Semestern aus einfachsten Programmen immer komplexere Befehlsfolgen zu konstruieren. Aus dieser Perspektive scheint sich das spätere Programmieren im Beruf lediglich durch die Anzahl der »lines of code« von dem Erlernten zu unterscheiden. Die mit diesem didaktischen Vorgehen den Auszubildenden **implizit vermittelte Problemlösungsstrategie** zeichnet sich durch das Vertrauen auf die individuelle Intuition aus und versucht, im »trial and error«-Verfahren den richtigen Lösungsweg zu finden.

Dieses Herangehen hat insbesondere in den Praktika, die während der Ausbildung abgeleistet werden, zu Problemen geführt. Die in der Ausbildung vermittelten Programmierkenntnisse waren in der beruflichen Realität **nur bedingt anschlußfähig**. Zwei Probleme wurden von den Dozenten, die die Schüler während der Praktikumsphasen betreuten, beobachtet:

a) Das implizit angelegte, rein intuitive Problemlösungsverhalten ist für die Entwicklung komplexer Software-Systeme nicht ausreichend. Geplantes, schrittweises Vorgehen muß erlernt werden, insbesondere sollte eine analytische Durchdringung des Problems am Anfang des Projekts stehen, bevor nach Lösungen gesucht wird. Großer Wert wurde daher darauf gelegt, statt des unsystematischen »trial and error« die Notwendigkeit eines geplanten Vorgehens zu vermitteln. "Dem Hackertum wurde der Kampf angesagt", so einer der Dozenten im

Expertengespräch. Das war die Geburtsstunde des Fachs »Software Engineering«.

b) Das implizit erlernte, rein intuitive Vorgehen bei der Problemlösung förderte weiterhin das individuelle »Bastlertum«. Dieses individualisierte Herangehen an die Lösung von Programmierproblemen blockierte in der Realität ein kompetentes Umgehen mit arbeitsteiligen Strukturen im Team. Da aber Software-Entwicklung heutzutage fast ausschließlich in kooperierenden Projektgruppen stattfindet und arbeitsteilig betrieben wird, fehlt den Schülern die Fähigkeit, ihre erworbenen Programmierkenntnisse (verknüpft mit sozialen Kompetenzen) erfolgreich in ein Software-Projekt einzubringen. Sie scheitern, weil sie nicht gelernt haben, Programmieraufgaben in Teamarbeit zu lösen. Insbesondere dort, wo die Schüler die Praktikumsprojekte nicht allein, sondern in kleinen Teams bewältigten, wurde dieses Problem deutlich erkennbar. In Reaktion auf diese Erfahrungen wurde das Fachgebiet »Software Engineering« inhaltlich so ausgelegt, daß die Auszubildenden lernten, Programmieraufgaben im Projekt zu zerlegen und arbeitsteilig erstellte Produkte aufeinander zu beziehen. Hierfür bot das klassische »Software Engineering« eine gute inhaltliche Grundlage.

Aus diesen Überlegungen entstand ein neues Fach, das inhaltlich an den Lehrauffassungen des »Software Engineering« orientiert war und der Vorbereitung auf die spätere berufliche Anforderungssituation und die Praktikumsphasen diente.

Heutzutage steht das Fach an einem Wendepunkt und vor der Notwendigkeit einer grundlegenden Innovation, um seine wichtige Vermittlungsfunktion weiterhin erfüllen zu können. Damit die Kompetenzen der Auszubildenden bezüglich einer sich grundlegend verändernden Praxis weiterhin anschlußfähig bleiben, muß nach unserer Auffassung das Themengebiet »Software Engineering« als **zentrales didaktisches Scharnier** zwischen den in der Ausbildung vermittelten Qualifikationen und der modernen beruflichen Anforderungssituation weiter ausgebaut und inhaltlich auf ein neues Leitbild der Software-Entwicklung orientiert werden. In seiner jetzigen inhaltlichen und didaktischen Ausgestaltung kann es diese Scharnierfunktion nicht erfüllen.

Software Engineering vor neuen inhaltlichen Anforderungen

Die Entwicklung von Software und der Aufbau unternehmensweiter IT-Systeme befindet sich im Wandel. In Anwenderunternehmen setzen sich zunehmend neue Formen und ein veränderter Einsatz der Informationstechnik durch. Die Veränderungen bewegen sich auf drei Ebenen:

- Informationstechnik,

- Arbeitsorganisation,
- Nutzungskonzepte.

In der **Informationstechnik** vollzieht sich eine rasante Entwicklung. CSA, Multimedia, Window-Technik, Objektorientierung, neue Programmiersprachen, immer kleinere und schnellere Prozessoren, Speichermedien mit immer höherer Speicherkapazität und Flexibilität und die zunehmende betriebliche und überbetriebliche Vernetzung verschiedenster Kommunikationsmedien eröffnen Möglichkeiten, die heute noch nicht vollends abzusehen sind. Parallel zu dieser technologischen Entwicklung erfolgt in den Unternehmen gegenwärtig eine **Reorganisation** der Strukturen und Abläufe. Unter dem Stichwort Geschäftsprozeß-Reengineering wird die spartenorientierte Organisation durch die Orientierung des Unternehmensaufbaus und der Abläufe an den Geschäftsprozessen der Produktgruppen ersetzt. Der Einsatz von Informationssystemen orientiert sich also zunehmend auf die Unterstützung ganzer Vorgangsketten von der Auftragserteilung bis zur Auslieferung und Fakturierung. Sowohl die technologische Entwicklung als auch der sich verändernde Einsatz der Informationstechnik in neuen Formen der Arbeitsorganisation erfordert neue **Nutzungskonzepte** für Computer und Software. Das Informationssystem wird als unternehmensweite Infrastruktur für Kommunikation und Kooperation genutzt. Die Software findet in diesen Systemen neue Einsatzmöglichkeiten als Werkzeug, das Arbeitsabläufe v.a. unterstützen und nicht mehr ersetzen soll.

Software-Entwicklung heute

In Folge der aktuellen Entwicklungen der Technik, der Arbeitsorganisation in den Unternehmen und der Nutzungskonzepte für die IT verändern sich **Charakter und Verlauf von Systementwicklungsprozessen**.

- **Integration von Arbeitsgestaltung und Software-Entwicklung**

Die Interdependenz moderner Formen der Arbeitsorganisation mit dem Einsatz und der Form moderner Informationssysteme macht eine weitgehende Integration der Arbeitsgestaltung und der Entwicklung von Software erforderlich.

- **Evolutionäre Entwicklungsprozesse**

Software und Arbeit werden evolutionär gestaltet. Die Software »wächst« in der Auseinandersetzung zwischen den technischen Möglichkeiten und der Neugestaltung der Arbeit bzw. des Nutzungskontexts. Anforderungen an die Software werden über den gesamten Entwicklungsprozeß hinweg reformuliert und über ein zyklisches Vorgehen implementiert.

- **Kooperative Software-Entwicklung**

Veränderungen des Nutzungskontexts und der Arbeitsorganisation machen eine Entwicklungspartnerschaft zwischen den beteiligten Fachabteilungen und der Informatikabteilung sowie zwischen Anwendern und Entwicklern im Sinne einer kooperativen Software-Entwicklung notwendig.

Rahmencurriculum »Moderne Software-Entwicklung«

Als Fazit dieser Analyse moderner Software-Entwicklung gilt:

- Die Veränderungen sind grundlegender Natur, Software-Entwickler stehen heute vor grundsätzlich neuen Aufgaben.
- Der Rahmen des Software Engineering in der Boehmschen Tradition wird durch diese Veränderungen gesprengt.

Das Fach »Software Engineering« muß heute die Anforderungen moderner Software-Entwicklung aufgreifen, neue Qualifizierungsziele bestimmen und die Inhalte des Fachs weiterentwickeln.

Das hier vorgestellte Rahmencurriculum »Moderne Software-Entwicklung« ist ein Beitrag zur inhaltlichen und didaktischen Neuorientierung des Fachs. Es soll Dozenten bei der Weiterentwicklung ihres Fachs unterstützen und Ausbildungsinstitutionen eine Basis für die konzeptionelle Neugestaltung der Ausbildungsgänge von IT-Fachkräften bieten.

In der Definition von Qualifizierungszielen und Qualifizierungsinhalten werden die aktuellen Veränderungen in der Software-Entwicklung aufgegriffen. Anhand einer wissenschaftlich fundierten Analyse der Basistrends werden inhaltliche Schwerpunkte vorgestellt. Das Rahmencurriculum orientiert sich an zwei zentralen Qualifizierungszielen:

1. Vermittlung eines modernen Aufgabenverständnisses

Die Veränderungen in der Software-Entwicklung fordern von IT-Fachkräften ein ganzheitliches Verständnis ihrer beruflichen Aufgaben:

- Software-Entwicklung ist die Entwicklung von Software im Kontext der Gestaltung von Arbeit.
- Software-Entwicklung ist eine kooperative Aufgabe.
- Software-Entwicklung ist ein komplexer sozialer Prozeß.

2. Vermittlung einer modernen Handlungsorientierung »Software-Entwicklung als kooperative Arbeitsgestaltung«

Software-Entwicklung fordert von IT-Fachkräften die Kompetenzen

- Software-Entwicklung mit der Arbeitsgestaltung zu verknüpfen,
- mit Anwendern zu kooperieren und
- die Software-Entwicklung als sozialen Prozeß zu organisieren und zu steuern.

Das Rahmencurriculum »Moderne Software-Entwicklung« zielt auf die Vermittlung von Handlungsorientierungen, die diese Kompetenzen fördern und der Offenheit und Variabilität von Entscheidungssituationen gerecht werden. Die Wissensvermittlung zielt dabei im Wesentlichen auf das Erkennen von Entscheidungssituationen und der Vorstellung und Bewertung von Lösungsmöglichkeiten.

I Mensch - Arbeit - Computer

Die Arbeitswelt ist heute der wichtigste Einsatzbereich von Computern und Software. In vielen Unternehmen ist der Computer zu einem alltäglichen Werkzeug, oft aber auch zu einem alltäglichen Ärgernis geworden.

Software-Projekte führen zu tiefgreifenden Veränderungen in der Arbeitswelt. Mit der Entwicklung von Software übertragen IT-Fachkräfte bestimmte Anteile der Arbeit auf den Computer und gestalten somit auch die Arbeit des Menschen. Um dieser Verantwortung gerecht werden zu können, benötigen IT-Fachkräfte ein ganzheitliches Verständnis ihrer Aufgaben, das die Entwicklung von Software mit der Gestaltung von Arbeit verknüpft.

Ziel dieses Kapitels ist die Vermittlung eines ganzheitlichen Verständnisses von Software in ihrem Einsatz in der Arbeitswelt. Zentrale Aspekte von Arbeit werden in ihrem Verhältnis zu Einsatz und Form von Software dargestellt und sollen einen ersten Überblick über Gestaltungsanforderungen in der Software-Entwicklung verschaffen. Sie bieten in den folgenden Kapiteln die Grundlage für die Operationalisierung des Aspekts »Arbeitsgestaltung« und damit für die Entwicklung einer auf »Arbeitsgestaltung« ausgerichteten Handlungsorientierung.

1 Probleme der Software-Entwicklung

Mit der fortschreitenden Entwicklung im Bereich der Informationstechnik werden dem Computer immer weitere Einsatzgebiete und -formen erschlossen. Er ist bereits weit in unser Alltagsleben vorgedrungen. Computer werden heute überall eingesetzt. Kaum ein Lebensbereich ist davon ausgeschlossen. Er steht den Kunden einer Bank für die direkte Abwicklung ihrer Bankgeschäfte zur Verfügung. Die Fahrgäste von Bussen und Bahnen beziehen über ihn Fahrplaninformationen und ihre Fahrkarten. Im kulturellen Leben entstehen neue Kunstformen, so wird etwa die Musik durch die Möglichkeiten des Computers radikal verändert. Der Computer wird nicht nur für Kinder zum Spielpartner. Im politischen Bereich eröffnen Computer neue Möglichkeiten des Informationszugangs für Politiker und Bürger.

In diesem Potpourri neuer Einsatzformen blieb eine unerwähnt: der **Einsatz des Computers in der Arbeitswelt**. Im Gegensatz zu vielen anderen Einsatzbereichen des Computers besteht hier für die Beschäftigten eines Unternehmens häufig keine Möglichkeit mehr, sich für oder gegen den Einsatz des Computers zu entscheiden. Anders als in der Bank, in der jeder dem Sachbearbeiter am Schalter sein Anliegen vortragen und den Computer noch vermeiden kann, sind

viele Menschen im Arbeitsleben gezwungen, den Computer einzusetzen. Arbeit ist in diesen Bereichen schon voll auf den Computer eingestellt, sie ist ohne ihn kaum noch vorstellbar. Computer und Software in der Arbeitswelt bringen deshalb ein besonderes Maß an Verantwortlichkeit für IT-Fachkräfte mit sich.

Viele Unternehmen und ihre Beschäftigten sehen sich heute mit dem Computer konfrontiert; sowohl mit seinen Vorzügen als auch mit seinen Nachteilen. Längst nicht jede Anschaffung oder jedes Software-Projekt stellt sich am Ende als Erfolg heraus. Statt dessen stehen nicht genutzter oder nicht fertiggestellter Software oft immense Kosten gegenüber. Noch häufiger aber sehen sich Beschäftigte gezwungen, mit Software und Hardware zu arbeiten, die nach ergonomischen und arbeitswissenschaftlichen Kriterien nicht mehr den Anforderungen moderner Arbeitsplätze entsprechen.

Viele Software-Projekte haben zu so erheblichen Problemen geführt, daß bereits seit einiger Zeit von der **»Software-Krise«** gesprochen wird. Spillner konstatiert ein Andauern dieser Krise seit nunmehr 25 Jahren: "Hauptprobleme bei der Erstellung und dem Einsatz von Software sind damals wie heute:

- Die ausgelieferte Software ist zu fehlerhaft, ja teilweise sogar unbrauchbar; dies kann bis zur Ablehnung der Verwendung der entwickelten Programmsysteme führen.
- Die Anwender sind mit der Software nicht zufrieden, sie entspricht nicht ihren Anforderungen und läßt sich nur mit großem Aufwand ändern oder anpassen. Dies hat erhebliche Auswirkungen auf die Pflege- und Wartungsphase, die meist unerwartet aufwendig und mit umfänglichen Kosten verbunden ist.
- Die zeitliche und kostenmäßige Kalkulation einer Softwareerstellung ist nur sehr vage möglich und führt in vielen Fällen zu erheblichen Termin- und Budgetüberschreitungen. Die Software wird zu teuer." [Spillner 1994, S. 49]

Angesichts dieser Probleme bei der Entwicklung von Software-Systemen und den vielen teuren Projektruinen in den Unternehmen stellen wir mit Ernst Denert die Frage: **"Warum ist es so schwierig, Software zu machen?"** [Denert 1992, S. 4] Warum ist es trotz vieler Anstrengungen, die etwa mit der Etablierung des Software Engineering unternommen wurden, bisher nicht gelungen, diese Probleme befriedigend zu lösen?

Denert findet als Praktiker, Projektleiter und Autor eines Buches zur methodischen Projektabwicklung auf diese Frage vier Antworten:

1. "Softwaresysteme gehören zu den komplexesten Gebilden, die je von Menschen geschaffen wurden." [ebd.]

Software besteht aus einer oft riesigen Zahl von Bauelementen, Maschinenbefehlen und Datenspeichern, die in vielfältiger Weise untereinander verknüpft sind. Strukturen und Abläufe sind oft außerordentlich komplex und nicht mehr überschaubar. Sie können weder im vorhinein noch im nachhinein in allen Konsequenzen durchdacht werden.

2. "Menschen machen Projekte." [ebd.]

In Softwareprojekten werden eine Reihe menschlicher Unzulänglichkeiten besonders deutlich: die Unfähigkeit, komplexe Sachverhalte gedanklich voll zu durchdringen, der Mangel an guten Ideen, Unfähigkeit zur Teamarbeit, Unfähigkeit zum »Programmieren im Großen« sind nur einige Beispiele, die genannt werden können.

3. "Das richtige Projekt machen." [ebd., S. 5]

Unternehmen sind sich in den seltensten Fällen über ihre Anforderungen an die Software im Klaren, so daß die Anforderungsdefinition "das wohl heikelste Kapitel der Software ist" [ebd.]. »Das richtige Projekt zu machen« ist meist schwieriger als »das Projekt richtig zu machen«.

4. "Der Stand der Kunst der Methoden und Werkzeuge für die Softwareentwicklung läßt zu wünschen übrig." [ebd.]

Der entstehende Mangel soll dann häufig durch eine »Softwarebürokratie«, also durch den überzogenen Einsatz von Verfahrensvorschriften, Aktivitätenchecklisten und anderen formalen Mitteln kompensiert werden. Kreatives Arbeiten wird damit systematisch behindert.

Diese vier Ursachen für die vielfältigen Probleme in der Software-Entwicklung lassen sich auf die **Besonderheiten** zurückführen, **die das »Produkt« Software gegenüber vielen anderen technischen Produkten heraushebt**. Im Vergleich zu anderen technischen Produkten sind die Unterschiede wesentlich aussagekräftiger als die Gemeinsamkeiten. Neben der Komplexität, die für Denert die entscheidende Rolle spielt, unterscheidet sich für Balzert Software "(...) von anderen technischen Produkten prinzipiell in folgenden Punkten:

- Software ist ein immaterielles Produkt.
- Software altert nicht.
- Software unterliegt keinem Verschleiß und benötigt keine Wartung im üblichen Sinne.
- Ersatzteile im herkömmlichen Sinne, die bei auftretenden Fehlern eingesetzt werden, gibt es für Software nicht.

- Software ist i.allg. leichter und schneller änderbar als ein technisches Produkt. (...)
- Software ist kein Serienprodukt." [vgl. Balzert 1989, S. 3]

Ihre wichtigste Besonderheit aber liegt darin begründet, daß durch Software ein Anteil menschlicher Tätigkeiten maschinisiert wird, der bisher allein dem menschlichen Denken vorbehalten war: der Umgang mit Informationen. Analogien zur Entwicklung anderer technischer Produkte, die eher stoffliche oder energetische Aspekte menschlicher Tätigkeiten maschinisieren (etwa Flugzeuge oder Produktionsmaschinen), sind deshalb nur wenig aussagekräftig. Im Gegensatz zu diesen wird Software in Bereichen eingesetzt, in denen nicht physikalische, sondern soziale Gesetze bestimmend sind. Coy identifiziert deshalb die bisher **unzureichende Reflexion des Wechselspiels von technischer Gestaltung und sozialer Wirkung** von informationstechnischen Systemen als wesentliche Ursache der Softwarekrise. [Coy 1992, S. 17] [138]

Die Anforderungen an die Software-Entwicklung, die allzu häufig zu schwerwiegenden Problemen werden, sind zu großen Teilen dadurch bedingt, daß durch den Computer das Verhältnis des Menschen zu seiner Arbeit neu bestimmt wird. Da bestimmte Anteile der Arbeit dem Computer übertragen werden, verändert sich die Arbeit des Menschen oft nachhaltig. Erst mit der Gesamteinschätzung der erreichten Veränderungen kann das Projekt als Erfolg oder Mißerfolg eingestuft werden. »Nach mir die Sintflut« ist demgegenüber die Haltung vieler Software-Projekte, die ihre Aufgabe allein in der Konstruktion der Software sehen. Angesichts der zunehmenden Verstrickung von Arbeitstätigkeiten und Softwareeinsatz erscheint diese Haltung nicht mehr zeitgemäß und führt zu Problemen, die in den Unternehmen immer weniger akzeptiert werden.

Diese an der Arbeit von Menschen orientierte Beschreibung der Aufgabe »Software-Entwicklung« führt zu Anforderungen an die Arbeitsform, die ihr angemessen ist. Zunehmend wird Software in Projekten entwickelt, in denen Anwender und Entwickler zusammenarbeiten, um diesen engen Bezug zwischen den Erfordernissen der Arbeit und den Gestaltungsanforderungen an die Software zu realisieren. In Software-Projekten sollen also sehr unterschiedliche Menschen mit z.T. sehr unterschiedlichen Qualifikationen ein Team bilden und

---

[138] Auch für Fuchs-Kittowski ist die "(...) Gestaltung des Verhältnisses von technischem Automaten und schöpferisch tätigem Menschen, von formalem Modell und der nichtformalen natürlichen und gesellschaftlichen Umwelt" das philosophische, theoretische und methodologische Grundproblem der Informatik. [Fuchs-Kittowski 1992, S. 71] "Das kritiklose Akzeptieren der Welt des technischen Automaten als Modell für die gesamte Wirklichkeit erweist sich zunehmend als gefährlicher Irrtum." [ebd.]

gemeinsam mehr leisten, als die Summe der Einzelarbeiten der Teammitglieder je ergeben könnte. Mit der Frage »Wer sollte wie zusammenarbeiten?« werden Anforderungen aufgeworfen, die in vielen Projekten keine hinreichende Beachtung finden und deshalb zu schwerwiegenden sozialen Problemen führen.

Gemessen an diesen Anforderungen hinsichtlich der Inhalte der Aufgabe »Software-Entwicklung« und der Arbeitsformen ist **der »state of the art« der Software-Entwicklung** an einem reduzierten Verständnis von Software-Projekten ausgerichtet. Durch die einseitige Orientierung auf die Software-Konstruktion und die technischen Anforderungen der Software-Entwicklung werden die vorherrschenden Methoden und Vorgehensweisen weder der sozialen Realität in Projekten noch den Aufgaben eines Software-Projekts gerecht.

Die skizzierten Defizite in der aktuellen Situation vieler Software-Projekte lassen darauf schließen, daß die Software-Krise zumindest in ihrer aktuellen Ausprägung ein Resultat der mangelnden Reflexion von Software im Kontext Arbeit und der unzureichenden Ableitung angemessener Aufgabenbestimmungen und Vorgehensweisen ist. Die Besonderheiten von Software, ihre nicht-stofflichen Eigenschaften, ihr Spannungsverhältnis zu ihrem sozialen Einsatzkontext bestimmen den Charakter der Software-Entwicklung, die Arbeitsform und die Wahl von Vorgehensweisen und Methoden nachhaltig. Deshalb wollen wir im folgenden zunächst der Frage

- Welche Bedeutung hat Software im Kontext Arbeit?

nachgehen, um ihre Eigenschaften und ihr Verhältnis zur Arbeit des Menschen genauer bestimmen zu können. Diese Ergebnisse werden wir dann nutzen, um in den weiteren Kapiteln folgenden Fragen nachzugehen:

- Was kennzeichnet Software Projekte?
- Was ist der Charakter der Aufgabe Software-Entwicklung?
- Welche Vorgehensweisen und Methoden unterstützen diese Aufgabe?

2 Software im Arbeitsprozeß

Wir wollen uns der Frage nach der Bedeutung von Software im Kontext der Arbeitswelt von den Zwecken ausgehend nähern, die mit ihrem Einsatz in der Arbeit verfolgt werden.

Im betrieblichen Umfeld dient Software dazu, **Arbeitsprozesse zu optimieren**. Durch ihren Einsatz sollen Kosten gesenkt, die Flexibilität erhöht, der Durchlauf beschleunigt oder der Ressourcenverbrauch reduziert werden. Diese Optimierung soll durch die Übertragung bestimmter formalisierbarer Anteile der Arbeit

auf den Computer geschehen. Vieles kann der Computer schneller und fehlerfreier ausführen; er kann Informationen einer weiteren Verarbeitung flexibler zugänglich machen als jedes andere Speichermedium und die mit ihm erzeugten Arbeitsergebnisse sind beliebig reproduzierbar und veränderbar.

Werden formalisierbare Anteile des Arbeitsprozesses in Form von Software auf den Computer übertragen, kann er Operationen ausführen, die vorher zu den Tätigkeiten eines Menschen gehörten. **Welche »Arbeitsteilung« zwischen Mensch und Computer** allerdings wirklich zu einer Optimierung des Arbeitsprozesses führt, ist damit noch nicht geklärt. Bisher wurde häufig ein Vorgehen gewählt, nach dem alles, was formalisierbar ist, auf den Computer übertragen wurde. Es entstanden damit sehr große Software-Systeme, die ganze Arbeitsabläufe und die in ihnen enthaltenen Entscheidungssituationen dem Computer überließen. Häufig aber mußte festgestellt werden, daß auch eine unter technischen Gesichtspunkten einwandfreie Software zu katastrophalen Ergebnissen führte, wenn eine effektive Organisation des Arbeitsprozesses mit der fertiggestellten Software nicht möglich war.

Heute wird deshalb häufig schon von diesem globalen Anspruch der Abbildung abgesehen und die Entscheidung hinsichtlich der Funktionsteilung zwischen Mensch und Computer nach praktikablen Gesichtspunkten getroffen. Im Idealfall soll die Software hier zu einem alltäglichen Gegenstand werden, der die Arbeit unterstützt und selbstverständlich, ohne großen Aufwand oder langes Überlegen eingesetzt werden kann. Weist Software diese Eigenschaften auf, verbessert sie die Bedingungen zur Effektivierung des Arbeitsprozesses. **Software als »alltäglicher Gegenstand«** [Züllighoven 1992, S. 143] sollte deshalb Entwicklungsziel jedes Software-Projekts sein.

Die Entwicklung von Software als »alltäglichem Gegenstand« setzt aber ein Verständnis von Arbeit voraus, das einen begründeten und nachvollziehbaren Zusammenhang zwischen dem menschlichen Handeln und der Software herstellt. Wir suchen deshalb im folgenden nach den grundlegenden Bestimmungsmomenten und Perspektiven der Arbeit, um nachzuvollziehen, was die Arbeit in der aktuellen Situation der Unternehmen, in denen heute Software-Projekte durchgeführt werden, bestimmt.

2.1 Die allgemeinen Bestimmungsmomente der Arbeit

Wir finden Arbeit in so vielen Erscheinungsformen, daß es auf den ersten Blick schwer fällt, Gemeinsamkeiten zu erkennen und Kategorien zu entwickeln, die für alle Formen von Arbeit anzuwenden sind. Arbeit ist die Arbeit am Fließband in

der Fabrik, die Arbeit des Sachbearbeiters in der Verwaltung, die Arbeit von Managern, Wissenschaftlern, Lehrern usw.

Trotz aller Unterschiede kann Arbeit aber in allgemeiner Form beschrieben und auf ein Kategoriensystem reduziert werden, das alle ihre Aspekte umfaßt und so den Grundstein für die Analyse aller denkbaren Formen von Arbeit legt.[139]

Was also macht Arbeit in ihrer allgemeinen Form aus? Betrachten wir Arbeitsprozesse gleich welcher Art, so fällt zunächst auf:

Arbeit ist eine leibliche Aktivität. Wir Menschen sind lebendige, sinnliche Wesen, "die in einer natürlichen Welt leben, die wir mit unseren Sinnen erfassen und mit unseren natürlich-leiblichen Mitteln beeinflussen können." [Raeithel 1992, S. 128] Je nachdem, mit welchen Aspekten der Umwelt sich ein Mensch auseinandersetzt, unterscheidet sie sich ihrer Form nach. Der Philosoph denkt. Der Bäcker knetet den Teig mit seinen Händen. Der Angestellte im Büro führt mit der Hand die notwendigen Bewegungen aus, um eine Tabelle auszufüllen. Jede Arbeitstätigkeit ist mit einer Aktivität des Leibes verbunden.

In der leiblichen Aktivität nimmt der Mensch seine Umwelt sinnlich wahr und verändert sie. Er fühlt die Beschaffenheit des bearbeiteten Materials. Er sieht, ob das Resultat der Arbeit den gesetzten Zielen entspricht. Er sieht, hört oder fühlt Fehler im Arbeitsprozeß. Er stellt seine weitere Aktivität auf die Ergebnisse dieser Wahrnehmungen ein.

Arbeit ist Arbeit mit Dingen und Gegenständen unserer Umwelt. In der Arbeit werden Werkzeuge und Maschinen eingesetzt, mit denen Rohstoffe oder Zwischenprodukte bearbeitet werden. In der industriellen Produktion ist dies nicht anders vorstellbar, aber auch Theoretiker kommen nicht ohne Werkzeuge aus. Ein Bleistift oder andere Zeichenwerkzeuge sind in jedem Fall notwendig, um Gedanken zu Papier zu bringen, komplexe Berechnungen anzustellen oder Modelle zu entwickeln.

Arbeit formt Gegenstände und wird selbst durch Gegenstände geformt. Der Mensch stellt in seiner Arbeit Gegenstände her, die für den Verbrauch, den Konsum oder für den weiteren Einsatz in Arbeitsprozessen vorgesehen sind. Diese Gegenstände verändern seine Umwelt. Damit werden die Ausgangsbedingungen für die weitere Arbeit immer wieder verändert und erweitert. Jede so

---

[139] In der folgenden Beschreibung von Arbeit folgen wir Arne Raeithel [Raeithel 1992, S. 128ff.], der die folgenden Merkmale allgemein für gegenständliche Tätigkeiten entwickelt hat.

geschaffene Situation birgt neue Probleme und Möglichkeiten, die durch die Arbeit bewältigt werden müssen. So ist die Arbeit heute z.B. durch die selbst geschaffenen ökologischen Probleme einerseits, aber auch durch die mit der Informationstechnik immer größer werdenden technischen Möglichkeiten andererseits geprägt.

Die Arbeit kann selbst zum Gegenstand von Arbeit werden. Die Arbeit wird nicht nur durch die sie umgebenden Gegenstände geformt, sondern auch durch den Menschen selbst. Arbeit wird reflektiert und neue Möglichkeiten werden entwickelt. "Dabei werden die Regelmäßigkeiten und funktionalen Muster von verallgemeinerbaren Handlungsformen zum Gegenstand der vorausgreifenden, antizipativen Form der Arbeit." [Raeithel 1992, S. 132] Dies geschieht tagtäglich und ungeplant, integriert in Arbeitsprozesse. In Organisationsabteilungen und in eigens dafür vorgesehenen Projekten wird die Gestaltung von Arbeit zur Hauptaufgabe von Menschen.

Arbeit existiert nur als Kooperation. In der Arbeit gehen Menschen Beziehungen untereinander ein, um Arbeitsergebnisse auszutauschen oder um gemeinsam Arbeitsergebnisse zu erzielen. Der Mensch ist in seiner Arbeit in gewachsene Strukturen (wie Organisationsformen, Kommunikationsbeziehungen, etablierte Arbeitsformen) eingebunden, die weitgehend unabhängig von ihm existieren und die ihn selbst und seine Arbeit formen.

Ausgehend von Arbeit als leiblicher Aktivität bis hin zur Kooperation ist Arbeit so in ihrer allgemeinen Form beschrieben. Auf dieser Basis identifizieren wir drei wesentliche **Bestimmungsmomente des Arbeitsprozesses**, von denen jedes eine eigenständige Perspektive auf ihn eröffnet.

- **Die Arbeitstätigkeit**

  "Im Arbeitsprozeß bewirkt also die Tätigkeit des Menschen durch das Arbeitsmittel eine von vornherein bezweckte Veränderung des Arbeitsgegenstandes." [Marx, MEW 23, S. 195]

- **Die Arbeitsmittel und -gegenstände**

  In der alltäglichen Arbeit geht der Mensch mit den ihm vertrauten Dingen selbstverständlich um. Diese Dinge sind für ihn Arbeitsmittel (wie etwa Werkzeuge und Maschinen) oder Arbeitsgegenstände (Materialien). Arbeitsmittel und Arbeitsgegenstände sind selbst bereits Ergebnisse von Arbeitsprozessen und werden nun zur Erfüllung neuer Zwecke eingesetzt. Sie bestimmen durch ihre Beschaffenheit und ihre Funktionalität Arbeitstätigkeiten und Kooperationsbeziehungen.

- **Die Kooperation**

  Im Arbeitsprozeß gehen Menschen untereinander bestimmte Verhältnisse zur Erreichung eines gemeinsamen Ergebnisses ein. Diese bezeichnen wir als Kooperation. In dieser Zusammenarbeit etablieren sich Beziehungen und Strukturen, die die Arbeitstätigkeit des einzelnen und schließlich auch die Form und Bedeutung von Arbeitsmitteln und -gegenständen maßgeblich mitbestimmen.

Diese drei allgemeinen Bestimmungsmomente des Arbeitsprozesses stehen nicht isoliert nebeneinander. Vielmehr sind sie in vielfältiger Weise untereinander verknüpft und gegenseitig bedingt. In der Arbeit des Menschen verschmelzen Arbeitstätigkeiten, Arbeitsmittel und -gegenstände und die Kooperation zu einem zweckgerichteten Ganzen. Die **Zwecke der Arbeit** bilden damit den Bezugspunkt für Form und Ablauf von Tätigkeiten, für die Form von Arbeitsmitteln und -gegenständen und die Form der Zusammenarbeit von Menschen. Die Beziehung der drei Momente ergibt sich daraus,

- daß der Mensch in seiner Arbeit bestimmte Zwecke und Ziele verfolgt, an denen er seine Handlungen ausrichtet und nach deren Maßgabe er Arbeitsmittel und -gegenstände einsetzt,
- daß Arbeitsgegenstände wie Werkzeuge so gestaltet werden, daß mit ihnen die verfolgten Zwecke erreicht werden können, indem sie an die Arbeitsverfahren, die eingesetzten Materialien und die gewünschten Eigenschaften der Arbeitsergebnisse angepaßt werden,
- daß die Zusammenarbeit von Menschen immer an der Verfolgung gemeinsamer Zwecke orientiert ist und sich nach dieser Maßgabe die Arbeitsteilung, die gemeinsame Nutzung von Werkzeugen und Materialien und die Form der Koordination der Arbeit ergibt.

Die Zwecke der Arbeit sind v.a. dadurch begründet, daß die Arbeit des Menschen im weitesten Sinne immer der Aufrechterhaltung der menschlichen Existenz dient. Wir arbeiten, um unsere Lebensgrundlagen zu erhalten und im gesellschaftlichen Produktionsprozeß unseren Beitrag dazu zu leisten. Die arbeitsteilige Gesellschaft führt dabei zu einer sehr weitgehenden Differenzierung dieser Zwecke, die ihren Charakter oft verdeckt. Sie verbergen sich hinter oft scheinbar willkürlichen Arbeitsaufträgen, die in der Arbeit an uns gerichtet werden.

Eine wesentliche Grundlage der Zwecksetzung, die wir in unserer alltäglichen Arbeit vornehmen, sind die Erfahrungen mit den zur Verfügung stehenden Werkzeugen und Materialien. Die Beschaffenheit des zu bearbeitenden Materials und die Beschaffenheit und Funktionalität der zur Verfügung stehenden

Werkzeuge vermitteln uns einen Eindruck von den in ihnen liegenden Möglichkeiten. So formulieren wir etwa unsere Ansprüche an Form und Qualität unserer Arbeitsergebnisse - z.B. in der Gestaltung einer Tabelle, Statistik, Grafik - immer in Abhängigkeit von den Werkzeugen, die uns zur Verfügung stehen. Den Zweck in Form des ideellen Arbeitsergebnisses bestimmen wir danach, ob uns etwa nur Papier und Bleistift oder Software zur Verfügung stehen, danach ob die Software komfortabel ist oder nur eine geringe Funktionalität aufweist.

Darüber hinaus wird die Zwecksetzung wesentlich durch die Kooperationsbeziehungen, in denen wir uns befinden, bestimmt. Je nachdem, wie gut wir mit Kollegen zusammenarbeiten, wie weitgehend wir mit ihnen gemeinsame Ziele verfolgen und uns über die Arbeitsteilung, Arbeitsformen, Termine oder Qualitätsstandards verständigen können, werden wir die Arbeitsergebnisse, an denen wir gemeinsam arbeiten, definieren.

Nach dieser sehr allgemeinen Bestimmung des Arbeitsprozesses läßt sich die **Rolle der Software in der Arbeit** genauer bestimmen. Als »alltäglicher Gegenstand« ist sie

- **Arbeitsmittel**, das in der Arbeit eingesetzt wird, um Informationen zu bearbeiten, oder
- **Arbeitsgegenstand** in Form von Informationen, die etwa in Datenbanken gespeichert sind und in das Arbeitsergebnis einfließen.

Als Arbeitsmittel bzw. -gegenstand ist Software über die in der Arbeit verfolgten Zwecke eng mit den Arbeitstätigkeiten und der Form der Kooperation verknüpft.

Jeder Arbeitsprozeß im Unternehmen, der zum Gegenstand eines Software-Projekts wird, weist die dargestellten allgemeinen Bestimmungsmomente der Arbeit auf. Ein Projekt, das sie in ihrer Einheit erfaßt und im Blick behält, läuft weniger Gefahr, wesentliche Aspekte der Gestaltung von Arbeit und der Anforderungen an die Software zu übersehen.

2.1.1 Arbeitsgegenstand Information

Arbeitsgegenstände sind diejenigen unserer Arbeit, die in das Arbeitsergebnis einfließen. In der industriellen Produktion, etwa im Automobilbau, sind das z.B. Rohstoffe oder Zwischenprodukte, die verarbeitet oder nur noch montiert werden. Sie gehen in ihrer spezifischen Beschaffenheit in den Arbeitsprozeß ein und bilden so Rahmenbedingungen, auf die sich der Arbeitsprozeß einrichten muß: Die eingesetzten Werkzeuge und Arbeitsverfahren müssen grundsätzlich zur Bearbeitung dieser Arbeitsgegenstände geeignet sein.

Ein großer Teil der Arbeitsergebnisse der Verwaltung, sei es in Produktionsunternehmen, Dienstleistungsunternehmen oder öffentlichen Verwaltungen, sind Informationen. Informationen entstehen einerseits im Produktionsprozeß und werden andererseits wieder für die Planung, Steuerung und Kontrolle des Produktionsprozesses eingesetzt. Sie werden in Form von Statistiken, Zeichnungen, Anweisungen, Plänen und vielem mehr hergestellt. Diese Form der Arbeit nimmt mittlerweile einen so großen Anteil der gesamtgesellschaftlich geleisteten Arbeit ein, daß vielerorts bereits vom Übergang zur Informationsgesellschaft gesprochen wird, deren Grundlage nicht mehr die Produktion materieller Güter sondern die von Informationen sei.

Informationen sind der wichtigste Arbeitsgegenstand von Verwaltungstätigkeiten. Produktionsdaten, Buchungsdaten, Kostenrechnungen, Umsatzzahlen u.v.m. werden in der Verwaltung der Unternehmen tagtäglich erzeugt und bearbeitet und damit als Grundlage für die Produktion bereitgestellt. Die zugrundeliegenden Daten liegen in der Regel in unüberschaubarer Fülle vor, so daß die zentrale Leistung der Verwaltung in einer Selektion und Verdichtung dieser Daten zu aussagekräftigen Informationen zu sehen ist.

Daten und Informationen sind für Verwaltungsprozesse Arbeitsgegenstände, die bearbeitet werden wie die Arbeitsgegenstände materieller Produktionsprozesse auch. Und doch zeichnen sie sich durch Besonderheiten aus, die etwa dazu geführt haben, daß die Bearbeitung von Informationen durch Maschinen erst in jüngster Zeit möglich ist und davor dem Kopf des Menschen vorbehalten war.

Information erfüllt in unserem Leben und in allen biologischen Prozessen eine zentrale Funktion. Wiener (ein Kybernetiker) sagte**: "Information ist Information, weder Materie noch Energie."** [Wiener 1948, S. 155] und stellte die »Information« als dritten elementaren Aspekt neben die Grundkategorien »Stoff« und »Energie«. In dieser Sichtweise haftet die Information den Dingen und Lebewesen unserer Umwelt an wie ihre stofflichen und energetischen Eigenschaften. Als solche ist sie die wesentliche Bedingung unseres Verhaltens und des Verhaltens aller lebenden Strukturen und Systeme auf der Basis energetischer und materieller Prozesse. [vgl. Seetzen 1992, S. 88] In unserem Alltagsverständnis bedeutet das, daß der Begriff »Information« dazu dient, "(...) das für einen bestimmten Sachverhalt, eine Person, ein Ereignis, einen Vorgang o.ä. relevante Wissen von dem in dieser Hinsicht unwichtigen zu unterscheiden (...)." [Franck 1990, S. 679] In diesem Sinne ist Information für uns ein zweckorientiertes Wissen, das wesentlichen Einfluß auf unsere Entscheidungen und Handlungen hat.

Eine offensichtliche Besonderheit besteht darin, daß Informationen nicht greifbar sind, denn sie bestehen nicht aus irgendeinem Stoff. Auch wenn sie auf ein Blatt Papier gedruckt oder elektronisch gespeichert sind, so ist diese Trägersubstanz doch offensichtlich völlig unrelevant für das, was die Information ausmacht. Was wir als Information ansehen, wird nicht durch die Trägersubstanz bestimmt, sondern gleichermaßen durch den Sender, mich als dem Empfänger und unserem Verhältnis zueinander. Dabei führt nicht so sehr die Frage nach der Struktur weiter, sondern vielmehr die Frage nach **Entstehung, Erhaltung und Nutzung von Informationen im Kontext menschlicher Arbeitsprozesse** . [vgl. Fuchs-Kittowski 1992, S. 72]

In der Kommunikationssituation wird Information nicht einfach durch Sprache übertragen. Sie entsteht erst im Verhältnis zwischen dem Gesagten und dem Vorverständnis des Hörers, des Empfängers. Der Austausch und die Entstehung von Informationen ist dabei an einen Kontext, einen Bedeutungsprozeß bzw. Sinnzusammenhang gebunden.[140] Zu denken ist etwa an Zahlen, denen oft an sich der Status einer Information zugebilligt wird. Erst die Verknüpfung der Zahl mit dem Begriff »Umsatz 1993« macht eine Zahl x zu einer Information, die in einem bestimmten sozialen Kontext entsteht und bearbeitet wird. Dabei ist dieser Kontext nicht auf die formalisierten Anteile, etwa der Buchhaltung, beschränkt. »Umsatz 1993« hat für den Manager eine andere Bedeutung als für den Sachbearbeiter und wird in Phasen sinkender Absatzzahlen mit anderen Interessen und Zwecken bearbeitet als in Phasen wirtschaftlicher Erfolge.

Was Information ist und welche Zwecke mit ihr verfolgt werden, wird im Unternehmen und in Abstufungen im Arbeitsprozeß definiert und unterliegt einer permanenten Entwicklung. Gerade der Einsatz von Computern und die Möglichkeit der Maschinisierung der Informationsverarbeitung erweitert den Raum möglicher Zwecksetzungen sprunghaft. So waren die Daten der Finanzbuchhaltung bisher oft nur der entsprechenden Abteilung im Rahmen der Finanzbuchhaltungsanwendung zugänglich. Als Grundlage von Planungs- und Entscheidungsprozessen standen sie nur in Form von Hardcopys zur Verfügung. Die größere Flexibilität moderner Software-Architekturen erlaubt nun auch den direkten Zugriff des Managements auf diese Daten und bietet neue Möglichkeiten der Entscheidungsunterstützung. Es entstehen neue Zwecke, die mit der Bearbeitung von Informationen verfolgt werden. Diese Möglichkeiten werden Schritt für Schritt im

---

[140] Fuchs-Kittowski vertritt dementsprechend die These: "Das Problem der Informationsentstehung ist von grundsätzlicher Bedeutung für die Modell- und Theoriebildung im Spannungsfeld von Computerprogramm und menschlichem Geist." [Fuchs-Kittowski 1992, S. 74]

Unternehmen realisiert und führen zu einer permanenten Neuformulierung der Ansprüche an den Computereinsatz.

2.1.2 Arbeitsmittel Software

Wir nehmen Informationen permanent auf, speichern und manipulieren sie. Diese informationellen Anteile der Arbeit haben sich bis zum Einsatz von Computern vor allem im Kopf abgespielt. In diesem Sinne sind sie als »Kopfarbeit« zu bezeichnen. [vgl. Nake 1992] Mit dem Computer werden nun auch diese Anteile der Arbeit der Maschinisierung zugänglich.[141] Dies ist eine einschneidende Veränderung unserer Arbeitswelt, denn bis vor wenigen Jahrzehnten noch war die »Kopfarbeit« allein dem Menschen vorbehalten, Werkzeuge und Maschinen wurden allein zu dem Zweck eingesetzt, die Handarbeit des Menschen zu unterstützen. [vgl. ebd., S. 185]

Um Informationen mit dem Computer ver- und bearbeiten zu können, muß Software zur Verfügung stehen, die einen Teil der zur Informationsverarbeitung notwendigen Operationen repräsentiert. **Software** wird damit zu einem Arbeitsmittel, mit dem der Arbeitsgegenstand »Information« bearbeitet werden kann. Je nach den im Arbeitsprozeß realisierten Konzepten der Arbeitsgestaltung begegnet uns Software dabei in sehr unterschiedlichen Formen. Wir finden Software, die wir in Analogie zu den Arbeitsmitteln in der Produktion als

- Automat,
- Maschine oder
- Werkzeug [vgl. Züllighoven 1992, S. 146]

bezeichnen können. Mit diesen Formen wird die Frage nach der Formalisierung von Arbeit sehr unterschiedlich beantwortet. Dies spiegelt sich in den differierenden Entscheidungen hinsichtlich Einfluß- und Kontrollmöglichkeiten und der Durchschaubarkeit der Informationsverarbeitung durch den Menschen wider.

---

[141] Maschinisierung verstehen wir als die Übertragung bestimmter Anteile der Arbeit auf eine Maschine. Die Möglichkeit der Maschinisierung der informationellen Anteile der Arbeit beschränken sich dabei auf die berechenbaren Operationen, die in algorithmischer Form beschreibbar sind. "Computer können berechenbare Funktionen auswerten. Handlungen, die als berechenbare Funktionen beschrieben sind, können mithin an Computer übergehen. Arbeitsvorgänge, die zwar nicht in Gänze berechenbar sind, in die aber berechenbare Operationen als Teile eingebettet sind, können als interaktive Benutzung des Computers organisiert werden." [Nake 1992, S. 196]

Viele Softwaresysteme verhalten sich wie **Automaten**. Ein Automat wird per Befehl oder Knopfdruck eingestellt und gestartet und führt eine mehr oder weniger lange Kette von Operationen mit dem Datenmaterial durch. Dieser Prozeß kann begleitend kaum noch kontrolliert und gesteuert werden, so daß erst anhand des Ergebnisses Fehler festgestellt werden können. Dies ist eine Form der Qualitätskontrolle, die nur noch für sehr wenige einfache Arbeitsprozesse sinnvoll erscheint.

Software als **Maschine** basiert auf der Teilung von Operationsketten durch Kontrollpunkte und formalisierte Entscheidungsmöglichkeiten. Die Durchführung der Operation ist also an definierten Punkten kontrollierbar und idealerweise auch revidierbar. Die Transparenz der Manipulation von Daten ist gegenüber der Software als Automat wesentlich erhöht. Nachteile dieser Form von Software sind v.a. in der starken Formalisierung von Entscheidungssituationen zu erkennen, die insbesondere bei sehr großen Software-Systemen zu einer Inflexibilität des Arbeitsprozesses führen.

Bei der Nutzung von Software als **Werkzeug** geht die Aktivität vom Menschen aus. [ebd., S. 145] Die Manipulation von Daten und Informationen geschieht weitgehend transparent, d.h. die Handhabung von Software-Werkzeugen erlaubt eine sinnliche Rückkopplung zwischen dem Material und dem angestrebten Zweck der Arbeit und ist somit die Basis für eine zeitnahe Kontrolle und Revision von Arbeitsergebnissen. Software als Werkzeug ist dem Leitbild der Unterstützung von Arbeit verpflichtet. Angestrebt ist die Bereitstellung flexibler Handlungsmöglichkeiten, die eine situationsabhängige Entscheidungsfindung erlaubt.

Software ist heute das zentrale Arbeitsmittel der Sachbearbeitung in Verwaltungen. Je nach den verfolgten Konzepten der Arbeitsgestaltung kann sie verschiedene Formen annehmen, in denen sich die Zwecke der Arbeit und ihr Verhältnis zur lebendigen Arbeit widerspiegelt. In ihrer Form und Beschaffenheit repräsentiert Software damit

- die Zwecke, die im Unternehmen und im Arbeitsprozeß verfolgt werden,
- das Verhältnis der zur Verfügung stehenden Informationen zu den angestrebten Informationen,
- die notwendigen Tätigkeiten zur Erstellung des Arbeitsergebnisses und
- die Kooperationsverhältnisse, in denen das Arbeitsergebnis erzielt werden soll.

2.2 Neue Unternehmensstrategien und Software-Einsatz

Software als Bestandteil eines umfassenden Konzepts von Arbeit muß heute vor dem Hintergrund sich wandelnder Unternehmensstrategien und der strategischen Einbettung der Informationstechnik verstanden und gestaltet werden.

Baethge und Oberbeck stellen für den Dienstleistungsbereich fest, daß die Unternehmen die Gestaltung neuer Markt- und Unternehmensstrukturen und die Realisierung neuer (informations-)technischer Konzepte integrieren. **Arbeitsprozesse und Organisationsformen werden neu konzipiert und mit dem Einsatz der Informationstechnik verknüpft**, um neue Unternehmensziele betrieblicher Effizienz und Flexibilität zu erreichen. [Baethge/Oberbeck 1990, S 150]

Bis zum Ende der 70er Jahre standen **quantitative Ziele** im Vordergrund, die durch einen punktuellen Einsatz der Datenverarbeitung erreicht werden sollten. In einer funktionsorientierten Unternehmensstruktur wurde die Datenverarbeitung in denjenigen Bereichen eingesetzt, in denen große Mengen von Daten mit vergleichsweise einfacher Datenstruktur erfaßt und in Arbeitsprozessen mit hohen Anteilen automatisierbarer Operationen bearbeitet werden sollten. Hilfs- und Routinefunktionen wurden im Zuge dieser Entwicklung von der Vorgangsbearbeitung abgespalten. [ebd., S 152] Durch diese Maßnahmen sollten eine Beschleunigung des innerbetrieblichen Informationsflusses, die Transparenz der innerbetrieblichen Arbeitsabläufe und eine Informationsbevorratung erreicht werden. [ebd., S 151]

Seit Ende der 70er Jahre sehen sich die Unternehmen mit einer anderen Marktsituation konfrontiert. Die quantitative Optimierung der Angebote reicht für den Erfolg am Markt nicht mehr aus, **qualitative Ziele** treten in den Vordergrund. Die bei den Sachbearbeitern verbliebenen Tätigkeiten sowie Planungs- und Steuerungsprozesse, aber auch die Kundenbeziehungen und Marktbedingungen werden verändert. **Mit Hilfe des Technikeinsatzes soll nun die Marktantizipation und Marktsteuerung verbessert werden**. [ebd., S 153] "Ein schnellerer Durchlauf von Kundenanfragen, eine Verminderung von Bearbeitungsfehlern, möglichst zeitaktuelle Auskünfte über Kontostände, Vertragsmodalitäten sowie über Produktions- und Lieferzeiten, all dies kann zur Steigerung der Dienstleistungsqualität und der Flexibilität gegenüber Kunden beitragen. Eine effizientere Organisation betriebsinterner Abläufe verspricht eine Erweiterung der Handlungspotentiale am Markt. Zur Disposition stehen damit jahrzehntelang erprobte Arbeitsteilungsmuster und Ablaufprozeduren (...)". [ebd., S 155]

So wird etwa in einigen Bereichen der Einsatz der IT mit integrierter Vorgangsbearbeitung verknüpft. Ehemals abgespaltene Arbeitsanteile werden integriert, die Arbeitsteilung wird zum Teil aufgehoben und so mehr Kundennähe und

Qualität erreicht. [ebd., S 156] Diese Rationalisierungsstrategie basiert dabei wesentlich auf dem "unauflöslichen Zusammenhang von Techniknutzung und qualifizierter Arbeitskraft", auf der aktiven "Gestaltung der Marktbeziehungen unter Hinzuziehung technischer Unterstützungs- und Steuerungsprogramme" statt auf dem bis dahin vor allem wirksamen Prinzip der Automation und Verdrängung der Arbeitskraft. [ebd.]

In vielen Unternehmen des Dienstleistungsbereichs zeichnen sich grundlegende Veränderungen ab. Die arbeitsorganisatorischen Strategien beziehen sich dabei auf ein hohes Niveau des Einsatzes von Informations- und Kommunikationstechnik, die neue Formen der Arbeitsteilung erst ermöglicht: "(...) dem organisatorischen Denken kann eine Prozeßorientierung zugrunde gelegt werden, und neue Möglichkeiten der Dezentralisierung in organisatorischer, räumlicher und geschäftlicher Hinsicht werden eröffnet." [Bullinger/Niemeier 1990, S. 81] In diesen neuen Formen der Arbeitsteilung setzen Rationalisierungsstrategien und Technikeinsatz auf Aufgabenintegration und -erweiterung für den einzelnen Sachbearbeiter und führen zu komplexen Tätigkeitsprofilen. [vgl. Baethge/Oberbeck 1990, S 157f.]

Die **enge Verknüpfung arbeitsorganisatorischer Innovationen mit dem Einsatz moderner Informationstechnik** führt dazu, daß Software-Projekte zu festen Bestandteilen dieser Innovationsprozesse werden. Im Software-Projekt wird mit der Gestaltung von Software der Arbeitsprozeß in seiner konkreten Form bestimmt. Arbeitsorganisatorische Ziele wie Aufgabenintegration, die Einführung von Gruppenarbeit, die Integration der Qualitätskontrolle in den Arbeitsprozeß u.a. müssen im Software-Projekt konkret realisiert werden, indem Software entwickelt wird, die diesen Anforderungen entspricht. Mit der Software als Arbeitsmittel oder -gegenstand müssen die projektierten Arbeitstätigkeiten und die angestrebten Kooperationsformen realisierbar sein.

3 Anforderungen an die Software-Entwicklung

Die **Software-Krise** resultiert in ihrer aktuellen Ausprägung darin, daß der Charakter von Software im Kontext der Arbeitswelt nicht hinreichend reflektiert und unangemessene Aufgabenbeschreibungen und Vorgehensweisen für Software-Projekte abgeleitet werden. Statt einer technikorientierten Sicht des Gestaltungsgegenstands »Software« benötigt das Software-Projekt, welches heute mit der Entwicklung von Anwendungssoftware betraut ist, ein Verständnis von Arbeit, das Software systematisch mit der lebendigen Arbeit des Menschen verknüpft.

Dieses **ganzheitliche Verständnis von Arbeit und Software** weist der Software die Rolle von Arbeitsmitteln und -gegenständen in einem zweckgerichteten

Arbeitsprozeß zu. Damit ist sie systematisch mit den Zwecken der Arbeit und der Gestaltung individueller Tätigkeiten und der Kooperation im Arbeitsprozeß verknüpft.

In Form von Informationen stellt Software einen zentralen **Arbeitsgegenstand** im Arbeitsprozeß dar, der sich gegenüber anderen Arbeitsgegenständen durch die Besonderheit seiner nicht-stofflichen Beschaffenheit auszeichnet. Informationen werden im sozialen Kontext des Arbeitsprozesses gemäß der Zwecke der Arbeit definiert und unterliegen damit einer prinzipiellen Variabilität. In Form einer Anwendung ist Software ein **Arbeitsmittel**, in dem die Zwecke des Arbeitsprozesses, die eingesetzten Verfahren und Vorgehensweisen vergegenständlicht sind. Damit kann sie je nach verfolgtem Konzept der Arbeitsgestaltung sehr unterschiedliche Formen annehmen.

Die Gestaltung von Arbeitsmitteln und -gegenständen ist ihrem Wesen nach kein Konstruktionsprozeß, in dem formale Vorgaben technisch umgesetzt werden. Sie ist vielmehr in einen Prozeß der Gestaltung von Arbeit eingebettet, in dem die Frage nach der Formalisierung bestimmter Anteile der Arbeit nach strategischen Gesichtspunkten getroffen wird. Deshalb muß **Software als Bestandteil eines umfassenden Konzepts von Arbeit** vor dem Hintergrund moderner Unternehmensstrategien verstanden und gestaltet werden.

In den Prozessen moderner Büroarbeit sind die Arbeitstätigkeiten, die Software, die in diesen Tätigkeiten eingesetzt wird, und die Kooperation als das Verhältnis, das alle am Arbeitsprozeß Beteiligten eingehen, weder gedanklich noch real zu trennen. Diesem Umstand muß die Entwicklung von Software Rechnung tragen. Software-Entwicklung ist demnach präziser als Arbeitsgestaltung zu bezeichnen und umfaßt gleichermaßen die Neugestaltung von Arbeitstätigkeiten und Kooperation sowie die Konstruktion von Software.

II Das Software-Projekt

In diesem Kapitel wird das Software-Projekt als diejenige Organisationsform beschrieben, in der Software-Entwicklung heute stattfindet. Für die IT-Fachkräfte ist das Software-Projekt der **Handlungsrahmen**, in dem sie sich kompetent bewegen müssen. Handlungskompetenz bedeutet hier, auf der Basis eines ganzheitlichen Verständnisses der Handlungssituation eigene Pläne und Strategien zur Bewältigung dieser beruflichen Anforderung zu entwickeln und diese dann erfolgreich umsetzen zu können. Dabei müssen die Handlungspläne der jeweiligen Spezifik des Projekts gerecht werden und nicht auf die Bewältigung einer vereinfachten Modellvorstellung orientiert sein. Dies erfordert eine analytische Durchdringung des Gegenstands durch die Lernenden.

Dieses Kapitel soll die Grundlagen für ein ganzheitliches Verständnis des Handlungsraums »Software-Projekt« legen. Zentrale Aspekte werden problemorientiert vorgestellt, um einen Überblick zu schaffen. Sie werden in den folgenden Kapiteln weiter vertieft. Deshalb hat dieses Kapitel für den Fortgang des gesamten Curriculums einleitenden Charakter.

1 Einführung

Software-Entwicklung findet in der Praxis im allgemeinen im Projekt statt, also im Rahmen einer zeitlich, personell und inhaltlich abgegrenzten organisatorischen Einheit. Die Besonderheit eines Projektes besteht vor allem darin, daß eine bestimmte Anzahl von Menschen für einen festgelegten Zeitraum mit einer definierbaren Aufgabe betraut wird. Anders als in herkömmlichen Organisationsformen, wie etwa Abteilungen, hat das Projekt ein definiertes Ende. Bei Erfüllung des Auftrags ist das Projekt beendet. Aus dieser Tatsache resultiert eine Spezifik von Projekten: Sie sind in der Art und Weise des Zustandekommens und der Abwicklung immer einmalig. Die DIN 69 901 kennzeichnet daher ein Projekt als "ein Vorhaben, das im wesentlichen durch seine Einmaligkeit der Bedingungen in ihrer Gesamtheit gekennzeichnet ist". [DIN 69 901]

Einmaligkeit bedeutet in der Praxis: Kein Projekt gleicht dem anderen. Auch wenn ein Auftrag in ähnlicher Form schon mehrmals bewältigt wurde, muß das Ziel und der Lösungsweg jeweils neu bestimmt und mit den Beteiligten ausgehandelt werden. Auch wenn die Mehrzahl der Projektbeteiligten schon in mehreren Projekten miteinander zusammenarbeiteten, muß sich die Projektgruppe unter einer veränderten Zielstellung neu organisieren. Hinzukommende Mitglieder müssen integriert und Arbeitsformen im Team immer wieder neu entwickelt

werden. Projekte machen aufgrund ihrer Einmaligkeit für alle Beteiligten eine ständige Neueinstellung notwendig. Projekte sind - so gesehen - die organisierte Veränderung.

Projekte werden eingesetzt, um eine abgegrenzte Innovation zu erreichen. Sie werden gegenüber einer schrittweisen Innovationstätigkeit dann ins Leben gerufen, wenn eine Veränderung »schubartig« vollzogen werden soll oder wenn zur Bewältigung dieser Innovation Spezialisten hinzugezogen werden müssen, die nicht der betreffenden Abteilung angehören. Für die Software-Entwicklung gilt in der Regel beides: Die Einführung einer technischen Neuerung wird meist »schubartig« durchgeführt, zumindest bis zu einem bestimmten Entwicklungsschritt wie z.B. einer Pilotanwendung. Computerspezialisten aus anderen Unternehmensbereichen oder von externen Firmen, die mit den Mitarbeitern der entsprechenden Abteilung gemeinsam das Projekt durchführen, müssen darüber hinaus meist hinzugezogen werden.

Für die Mitarbeiter der entsprechenden Abteilung und die einbezogenen Spezialisten hat das Projekt jeweils unterschiedliche Bedeutung. Aus der Perspektive der »Betroffenen«, also derer, deren Arbeit sich durch den Einsatz eines neuen Computersystems verändern soll, ist ein Projekt ein zeitlich abgrenzbarer »Innovationsschub«, der neben der alltäglichen Arbeit vollzogen wird und bewältigt werden muß. Das Projekt ist für sie meist die Ausnahme von der Regel.[142] Für die hinzugezogenen Entwickler ist das Durchführen von Projekten die Regel. Ihre Arbeit verläuft von einem Projekt zum nächsten. Die Durchführung von Neuerungen ist ihre Aufgabe und die alltägliche Arbeit der betroffenen Beschäftigten ihr Arbeitsgegenstand.

2 Software und Software-Qualität

Der Gegenstand eines Software-Projektes ist zunächst einmal Software. Zu unterscheiden sind allerdings zwei Arten von Software-Produkten, die jeweils

---

[142] Wenn man sich allerdings die Geschwindigkeit anschaut, mit der in den letzten Jahren eine Innovation in den Unternehmen auf die nächste folgt, wird deutlich, daß diese Ausnahmen von der Regel zunehmend zum ungeregelten Normalfall werden. Innovationen und normaler Arbeitsprozeß verschwimmen immer mehr. Projekte werden zur normalen Begleiterscheinung des Arbeitsprozesses und nur noch die Veränderung hat Bestand. Das Projekt als zeitlich definierte Organisationsform wird unter diesen Bedingungen zur Farce, die jeweilige Abschottung der sich ablösenden Projekte voneinander wirkt kontraproduktiv für den Erfahrungstransfer zwischen den Beteiligten. Zu Recht merken daher Weltz und Ortmann an, daß an die Stelle des Projektmanagements das Prozeßmanagement treten müsse. [Weltz/Ortmann 1992, S. 164ff.]

spezifische Anforderungen an das Software-Projekt stellen: Software kann **mittelbar** oder **unmittelbar** von Menschen genutzt werden. [vgl. Reisin 1994]

Mittelbar genutzte Software wird zur Steuerung von technischen Systemen eingesetzt. Satelliten, Verkehrsleitsysteme oder Produktionsstraßen werden vermittels dieser Software lediglich indirekt vom Menschen gesteuert. Diese Software, die nur über Schnittstellen zu technischen Komponenten verfügt, wird hier als **"technisch eingebettet"** [ebd.] charakterisiert. Die Funktionalität und die Schnittstellen der technisch eingebetteten Software sind in der Regel formal klar definiert. Diese Formalisierung richtet sich nach der Spezifik der jeweiligen Technik. Bei der Entwicklung dieser Software stehen technische Qualitätskriterien im Vordergrund. Es müssen zwar häufig Spezialisten für die entsprechenden technischen Verfahren und Abläufe herangezogen werden, weil das Lesen von Handbüchern für das notwendige Verständnis nicht ausreicht; das Hinzuziehen der späteren Anwender ist aber für die Entwicklung meist nicht erforderlich.

Demgegenüber wirkt von Menschen **unmittelbar genutzte**, also **"sozio-technisch eingebettete"** Software in aller Regel in Arbeitssystemen. Die Entwicklung sozio-technisch eingebetteter Software stellt den Regelfall kommerzieller Projekte dar. Sie wird zur Erfüllung konkreter Arbeitsaufgaben genutzt und beeinflußt häufig die Kooperations- und Kommunikationsbeziehungen der arbeitenden Menschen sowie das Verhältnis ganzer Unternehmen oder Unternehmensbereiche zu- und untereinander. Bei der Entwicklung sozio-technisch eingebetteter Software ist ihr Konstruktionsprozeß untrennbar mit der Gestaltung der Arbeit von Menschen verbunden. Die Erfordernisse und Besonderheiten des konkreten Arbeitsprozesses verschmelzen im Gestaltungsprozeß mit den Spezifika der Entwicklung des technischen Systems zu einem komplexen, vielschichtigen »Anforderungsgemisch«. Die Entwicklung unmittelbar genutzter Software stellt so besondere Anforderungen an die Entwickler und den Entwicklungsprozeß, weil ihre Herstellung stets in einem untrennbaren Zusammenhang zur Veränderung des betreffenden Arbeitsprozesses erfolgen muß.

Die Kriterien der **Software-Qualität** spiegeln den doppelten Bezug von Software (Technik und Arbeitssystem) wider. Neben Kriterien zur technischen Qualität wird in den letzten Jahren in zunehmendem Maße der Anwendungsbezug der Software als zentrales Qualitätskriterium in den Vordergrund gestellt. Beispielhaft seien hier die Kriterien aus der 1988 verabschiedeten DIN 66234, Teil 8[143], zur

---

[143] Einen guten Überblick über die Qualitätskriterien der DIN 66 234, Teil 8, bieten Lauter [1987] und Rödiger u.a. [1991]. Neben dieser DIN ist die VDI-Norm 5005 für die software-ergonomische Gestaltung von Bedeutung. Ein Überblick findet sich bei Ziegler [1993] und Becker/Paetau [1992]. Die Entwicklung von Normen wird seit eini-

Gestaltung von Dialogschnittstellen sowie die darauf aufbauenden Kriterien aus dem Projekt EVADIS [Oppermann u.a. 1988] dargestellt.

Als technische Qualitätskriterien gelten:

- Verständlichkeit (von Gesamtentwurf, Modulen, Programmteilen und Programmablauf)
- Änderbarkeit (strukturelle, technische und problemorientierte Änderbarkeit)
- Wiederverwendbarkeit (von Modulen oder Teilen des Systems)
- Korrektheit (definierte Ein- und Ausgaben, deterministisch)
- Ausfallsicherheit (Berücksichtigung des Versagens der technischen Umgebung durch Sicherung von Teilresultaten, Protokolle und Transaktionsmechanismen)
- Effizienz (optimale Ausnutzung von Ressourcen wie Speicherplatz etc.) [vgl. Mehl u.a. 1993, S. 51f.]

Die Erfüllung dieser technischen Qualitätskriterien ist in modernen Software-Projekten kein Selbstzweck. In der Praxis beziehen sie sich stets auf übergeordnete Qualitätskriterien, die sich auch aus der optimalen Einbindung eines Software-Systems in den jeweiligen Aufgabenbezug und der allgemeinen Forderung nach humaner Gestaltung von Arbeit ergeben.[144] Beispielhaft für die humane Gestaltung einer Dialogschnittstelle sind folgende allgemeine Kriterien zu nennen:

- Aufgabenangemessenheit
  Als aufgabenangemessen gilt ein Dialog dann, wenn der Benutzer bei der Ausführung unterstützt wird, ohne durch die Eigenschaften des Dialogsystems unnötig eingeschränkt zu werden.
- Erwartungskonformität

---

gen Jahren auf europäischer Ebene intensiviert. Diese Entwicklung bestimmt immer mehr die Dynamik auf nationaler Ebene. Gegenwärtig wird die ISO-Norm 9241, die die ergonomischen Anforderungen an die Bürotätigkeit mit Bildschirmgeräten behandelt, in nationales Recht umgesetzt. Sie fordert beispielsweise eine Analyse der Arbeitsplätze zur Verhinderung körperlicher und psychischer Belastungen der Mitarbeiter und fördert ihre Beteiligung an der Ausgestaltung des Arbeitsplatzes. [vgl. CW 3/94]

[144] Diese Qualitätskriterien haben präventiven Charakter. In ihrer Allgemeinheit beziehen sie sich in erster Linie auf übergeordnete Humankriterien, wie sie u.a. von Hacker, Volpert oder Ulich formuliert werden. Die hier als Bezugspunkt dienenden »Humankriterien« sind: Schädigungsfreiheit, Beeinträchtigungslosigkeit, Persönlichkeitsförderlichkeit und Zumutbarkeit. [vgl. Maaß 1993]

Als erwartungskonform gilt ein Dialog dann, wenn er den Erwartungen des Benutzers entspricht, die er sich in der bisherigen Arbeit, in einer Schulung, im Umgang mit Handbüchern oder dem Dialogsystem erworben hat.

- Selbstbeschreibungsfähigkeit
  Als selbstbeschreibungsfähig gilt ein Dialog dann, wenn dem Benutzer auf Verlangen Leistungsumfang und Einsatzzweck des Dialogschritts erläutert werden können, jeder Dialogschritt unmittelbar verständlich ist oder er auf Verlangen dem Schritt entsprechende Erläuterungen erhält.
- Fehlerrobustheit
  Als fehlerrobust gilt ein Dialog dann, wenn trotz fehlerhafter Eingabe das beabsichtigte Ergebnis mit minimalem Mehraufwand erreicht wird und dem Benutzer die Fehler zum Zwecke der Behebung verständlich gemacht werden.
- Steuerbarkeit
  Als steuerbar gilt ein Dialog dann, wenn der Benutzer die Geschwindigkeit des Ablaufs, Art und Umfang von Ein- und Ausgaben sowie Auswahl und Reihenfolge von Arbeitsmitteln beeinflussen kann. [vgl. Maaß 1993, S. 195]

Die hier dargestellten Qualitätskriterien verdeutlichen, daß Software zwei zentrale Bezugspunkte hat, eine technische Seite und eine zweite Bezugsebene, die eng mit der Verwendbarkeit der Software in einem Arbeitssystem korrespondiert. Insbesondere die Kriterien »Aufgabenangemessenheit« und »Erwartungskonformität« spiegeln den Bezug zum Anwendungszusammenhang wider. Die konkrete Ausgestaltung dieser Anforderung erfordert allerdings die Auseinandersetzung mit dem konkreten Arbeitssystem und kann nicht in Form allgemein gültiger Kriterien vorab formuliert werden. Dieser Gesichtspunkt kommt in den europäischen Richtlinien zum Ausdruck. Insbesondere die Umsetzung der ISO-Norm 9241, die die ergonomischen Anforderungen für Bürotätigkeiten mit Bildschirmgeräten regelt, erfordert u.a. die aktive Mitgestaltung der Anwender und eine Analyse der Arbeitsplätze. [vgl. CW 3/1994]

3 Arbeitsgestaltung als Ziel des Software-Projekts

Bevor in Unternehmen Software-Projekte ins Leben gerufen werden, geht dem meist eine längere Phase der Unzufriedenheit mit den bestehenden Verhältnissen voraus. Im Arbeitsprozeß werden Störungen ausgemacht, es wird nach Wegen gesucht, Arbeiten schneller, besser oder bequemer zu bewältigen. Auch wird bisweilen vom Einsatz einer bestimmten Technik eine Lösung für ein bestimmtes Problem erwartet. Die Faszination der Technik führt dann oft zur Suche

nach Verbesserungen des Arbeitsprozesses. Aus diesen Diskussionen reift so allmählich eine Projektidee heran.

Die Initiatoren für Software-Projekte finden sich in der Praxis nicht nur in den betroffenen Anwendungsabteilungen, sondern auch in der DV-Abteilung, der Geschäftsleitung oder sogar in externen Beratungs- und Herstellerfirmen. Typisch für die Entstehung von Projekten in Unternehmen sind folgende Konstellationen:

- Der Anstoß für die Initiierung eines Software-Projekts kommt häufig aus einer Fachabteilung. Hier erwarten sich die Verantwortlichen durch den Einsatz neuer Technik die Behebung von Störungen im Arbeitsprozeß, die Beschleunigung der Vorgänge oder die Reduzierung der Kosten.
- Von den Entwicklern der DV-Abteilung wird eine Idee für ein Vorhaben ausgearbeitet und der entsprechenden Fachabteilung vorgeschlagen.
- In der Geschäftsleitung erhofft man sich durch den Einsatz neuer Technik zusätzliche Rationalisierungsgewinne, erwartet neue strategische Handlungsmöglichkeiten oder ist einfach von einer neuen Technik fasziniert.
- Die Initiative kommt von einem externen Unternehmen (Berater, Softwarehaus oder Hardwarehersteller), oft gestützt und vermittelt durch einen internen »Promoter«. [vgl. Weltz/Ortmann 1992, S. 19ff.]

Wird dann ein Projekt ins Leben gerufen, so ist dieses stets auf die **Verbesserung oder Effektivierung eines Arbeitsprozesses** ausgerichtet. Ob es sich um ein neues Textverarbeitungsprogramm handelt, durch welches das Erstellen von Schriftgut mit höherer Qualität erleichtert werden soll oder um die Verknüpfung zweier Abteilungen durch ein gemeinsames, computergestütztes Kommunikationssystem, durch die man sich eine Verbesserung der Kooperationsbeziehung beider Abteilungen und eine Verkürzung der Durchlaufzeiten erhofft: Immer ist das Ziel des Software-Projekts die Gestaltung von Arbeit unter Einsatz von Software. Die konkrete Anwendungssituation ist der Ausgangspunkt, die zentrale Gestaltungsdimension und das entscheidende Erfolgskriterium des Software-Projektes:

- Das Projekt baut auf den Besonderheiten des konkreten Arbeitsprozesses auf; die Spezifik der jeweiligen Anwendungssituation ist also der Ausgangspunkt des Projektes.
- Die Besonderheiten der Arbeiten, die hier ausgeführt werden, begleiten das Projekt als zentrale Gestaltungsdimension über alle Phasen hinweg.

- Der Erfolg oder Mißerfolg des Projektes mißt sich letztlich daran, ob es gelungen ist, den Arbeitsprozeß im angestrebten Sinne zu verbessern.

Das übergeordnete Ziel »Arbeitsgestaltung« findet aber bei der Ausformulierung des Projektauftrages in der Praxis oft wenig Berücksichtigung. In der Realität heißt es meist nicht: »diese oder jene Abteilung soll schneller an bestimmte Informationen herankommen«; oder: »wir wollen die Erstellung von ansprechend gestalteten Briefen erleichtern«. Der Auftrag des Projekts enthält oft vielmehr eine technische Beschreibung des angestrebten Projektziels. Sein eigentliches Ziel - die Arbeitsgestaltung - verbirgt sich hinter technischen Umschreibungen.

Hinter dieser »Metamorphose« des Entwicklungsziels verbirgt sich meist ein Wechsel der an der Entscheidung beteiligten Personengruppen. Während die Anwender durchaus Auslöser des Projektes sind, haben sie bei der Definition des Projektauftrags meist kein Mitspracherecht mehr. [Weltz/Ortmann 1992, S. 24] In einer amerikanischen Untersuchung wird hervorgehoben, daß "die Leute, die über die Entwicklung von Software entscheiden, selten auch die sind, die sie tatsächlich nutzen". [zit.n. Weltz/Ortmann 1992, S. 22] In der Konsequenz ist laut Weltz und Ortmann folgendes zu beobachten: "Die fachlichen Probleme, die ursprünglich die ersten Anstöße zur Projektentstehung gegeben hatten, traten eher in den Hintergrund, technische Fragestellungen dominierten mehr und mehr." [Weltz/Ortmann 1992, S. 24]

Diese technische Umschreibung des Projektziels birgt für die Projektbeteiligten scheinbar eine Reihe von Vorteilen.

- Mit dem Einsatz einer neuen Software verbundene Veränderungen der Arbeit müssen nicht offen benannt werden. Damit verknüpfte Interessenkonflikte werden nicht offen ausgetragen, sondern hinter vermeintlich technischen Entscheidungen versteckt. Der Projektauftrag wird technisch umschrieben und scheint so eindeutiger formuliert. Das Projekt kann daher reibungsloser beginnen und kommt scheinbar schneller zu Ergebnissen.
- Ist der Gestaltungsgegenstand nur ein technisches Produkt, so lassen sich die notwendigen Entwicklungsschritte des Projektverlaufs trotz aller Komplexität eindeutig definieren. Das Software-Projekt erscheint dann als reibungsloser Ablauf der Phasen zur ingenieursmäßigen Erstellung eines technischen Produkts. Projekte zur Software-Entwicklung scheinen nach den gleichen Prinzipien plan- und durchführbar wie solche zum Bau von Brücken oder Häusern.

In der Praxis scheitern Software-Projekte aber gerade dann besonders häufig, wenn anstelle der eigentlichen Arbeitsgestaltungsziele technische Umschreibungen in den Vordergrund gestellt werden und der Prozeß selbst nicht als Pro-

zeß der Gestaltung von Arbeit, sondern als Technikgestaltungsprozeß organisiert wird. [vgl. Rauterberg 1993] Die Gründe hierfür liefert die oben erwähnte empirische Untersuchung zu Software-Projekten:

- Gerade wenn Konflikte nicht offen ausgetragen, sondern hinter vermeintlich technischen Entscheidungen versteckt werden, behindern sie oft schwelend den weiteren Fortgang des gesamten Projekts. Immer wieder reiben sich die Projektbeteiligten an diesen Fragen auf. Die Zeitlimitüberschreitung vieler Projekte ist auf diese verschleppten Konflikte zurückzuführen. [vgl. Weltz/Ortmann 1992, S. 129f.] Werden Projektaufträge lediglich technisch umschrieben, so sind sie für den weiteren Fortschritt der Projekte hinderlich. Sie bilden aufgrund ihrer Allgemeinheit und Unzulänglichkeit keine stabile Grundlage und müssen im Laufe des Projektes häufig revidiert werden. [vgl. ebd., S. 24f.]
- Auf dem Papier liefern die klassischen ingenieurswissenschaftlichen Ansätze des Software Engineering eine überzeugende Beschreibung des Softwareentwicklungsprozesses und bieten scheinbar eine stabile Grundlage für Software-Projekte. In der Praxis werden diese Ansätze aber weitgehend ignoriert, auch wenn »nach außen« so getan wird, als wären sie die Grundlage des Projektes. Mit Blick auf den Einsatz entsprechender Methoden in der Praxis heben Weltz und Ortmann hervor: "Der Beitrag normativer Ansätze, die zum Projektmanagement angeboten werden, war (...) äußerst gering. Praxis und »Wissenschaft« (...) ignorieren einander weitgehend. Dies dürfte nicht zuletzt auf die normative Ausrichtung des akademischen Softwareengineering zurückzuführen sein, die den Blick auf die Praxis versperrt und die Entwicklung von Modellkonzepten erleichtert, die offensichtlich an den Erfordernissen der Praxis vorbeigehen." [ebd., S. 170f.] Die Autoren verweisen darauf, daß "eine Entmythologisierung der Softwareentwicklung als rein »ingenieursmäßiger« Entwicklungsprozeß" [ebd., S. 130] notwendig sei, um Projekte erfolgreich bewältigen zu können.

4 "To program is to understand!" - Kooperation als Notwendigkeit

Eine empirische Untersuchung von Software-Projekten zeigt, daß der Anteil sozialer Aktivitäten wie Besprechungen, Interviews etc. in Projekten viermal mehr Zeit in Anspruch nimmt als das Programmieren. [Brodbeck/Sonnentag 1993] Dieses auf den ersten Blick vielleicht überraschende Ergebnis ist Ausdruck der Tatsache, daß moderne Projekte in erster Linie arbeitsteilig betrieben werden und der beständigen Kooperation bedürfen.

In Software-Projekten findet **Kooperation** zum einen zwischen Entwicklern statt, die jeweils unterschiedliche Probleme oder Bereiche einer Gesamtaufgabenstellung bearbeiten; zum anderen zwischen Entwicklern und Anwendern aus den betroffenen Fachabteilungen sowie zwischen Anwendern aus verschiedenen Unternehmensbereichen. Über diese Kooperationsbeziehungen hinaus kommen IT-Fachkräfte im Laufe der Projekte häufig mit verschiedenen Gruppen zusammen, die mittelbar oder sporadisch am Projekt beteiligt sind. Hierzu zählen z.B. der Betriebsrat, nicht direkt betroffene Anwender oder Mitglieder der Geschäftsführung.

Besondere Bedeutung in modernen Projekten kommt der Kooperationsbeziehung zwischen Entwicklern und Anwendern zu. Gerade in der Qualität der Beziehungen zwischen diesen beiden Gruppen liegt eine wesentliche Ursache für den Erfolg oder Mißerfolg von Projekten, so das Ergebnis der Untersuchung von Weltz und Ortmann zu Verlauf und Erfolgsbedingungen von Software-Projekten. [Weltz/Ortmann 1992, S 176ff.]

Moderne Projekte zur Entwicklung sozio-technisch eingebetteter Software sind auf allen Gestaltungsebenen - auch hinsichtlich der technischen Konstruktion und Realisierung - durch das Ziel der Neugestaltung der Arbeit bestimmt. Mit der Entwicklung neuer Software-Systeme werden nicht nur einzelne computergestützte Arbeitsmittel zur Verfügung gestellt, sondern Arbeitsaufgaben unter Verwendung von Software-Werkzeugen gezielt verändert. Durch den Einsatz der Informationstechnik werden weiterhin die Kooperations- und Kommunikationsbeziehungen neu gestaltet. Weil die Entwicklung moderner Software-Systeme so an der Veränderung von Arbeitsprozessen orientiert ist, ist das Verständnis der konkreten Anwendungssituation unabdingbar für die Software-Entwicklung. Nygaard hat diese prinzipielle Bedingung für die Entwicklung moderner Software in dem programmatischen Satz "To program ist to understand!" zum Ausdruck gebracht. [Nygaard 1986]

Dieses Verständnis der Anwendungssituation läßt sich nicht durch die Entwickler alleine herstellen. Selbst wenn sie über eine so breit angelegte Qualifikation verfügen, daß sie die entsprechenden fachlichen Probleme (z.B. betriebswirtschaftliche Fragen oder physikalische Abläufe) ausreichend durchdringen, fehlt ihnen dennoch die Kenntnis der konkreten Abläufe. Jedes Arbeitssystem ist spezifisch; die konkrete Arbeitssituation ist jeweils auch Ergebnis der sozialen Prozesse zwischen den dort arbeitenden Menschen. Das beim Entwickler unter Umständen vorhandene Fachwissen über die Anwendungsprobleme ist nur eine Voraussetzung zum Verständnis des konkreten Arbeitsprozesses. Das zur Gestaltung moderner Software notwendige konkrete Gestaltungswissen läßt sich nur in **Kooperation zwischen Anwendern und Entwicklern** gewinnen.

Reisin schlußfolgert für die Entwicklung von sozio-technisch eingebetteter Software: "Ist der Prozeß der Software-Entwicklung nicht vorrangig auf technische Qualitätsmerkmale des Software-Produkts, sondern gleichermaßen auf die aufgabenspezifische und psychosoziale Qualität der Arbeitstätigkeiten seiner Benutzung ausgerichtet, so ist Kooperation zwischen Entwicklern und Benutzern bei der Gestaltung zwingend geboten. Demnach muß die Organisationsform solcher Software-Projekte die Beteiligung der Benutzer vorsehen und die Kooperation zwischen ihnen und den Entwicklern bei der Durchführung der Software-Entwicklung geeignet unterstützen." [Reisin 1994]

Das für die Gestaltung notwendige Wissen, über welches die Mitglieder eines Software-Projekts verfügen müssen, bezieht sich nicht nur auf die Kenntnisse der konkreten Abläufe und Verfahren im jeweiligen Arbeitsprozeß. Notwendig ist vielmehr auch ein Bewußtsein hinsichtlich der Rückwirkungen der entwickelten technischen Werkzeuge auf die Arbeit. Unter diesem Blickwinkel sind das jeweilige Wissen und die jeweiligen Wissenslücken von Entwicklern und Anwendern komplementär:

- Anwender verfügen über Kenntnisse und Erfahrungen der fachlichen und sozialen Aspekte der konkreten Anwendungssituation. Hinsichtlich der Wirkung softwaregestützter Werkzeuge auf die Arbeit fehlen ihnen in der Regel die Erfahrungen.
- Entwickler verfügen über allgemeine Kenntnisse und Erfahrungen der Wirkung computergestützter Werkzeuge auf die Arbeit, haben aber meist keine Kenntnisse der konkreten Arbeitssituation in sachlicher und sozialer Hinsicht.

Dieses komplementäre Expertenwissen ist eine notwendige Voraussetzung für die Gestaltung von Arbeitsprozessen unter Einsatz von Software-Systemen. Bei den meisten Projekten reichen die beiderseits vorhandenen Erfahrungen und Kenntnisse nicht aus, um die Aufgabe zu lösen - selbst wenn es beiden Seiten gelingen sollte, geeignete Kommunikationsformen zum Austausch dieser Kompetenzen zu finden. Ausschlaggebend für den Projekterfolg ist vielmehr ein **gemeinsamer Lernprozeß beider Seiten** im Rahmen des Projekts. Neues Wissen muß in einem kooperativen Prozeß gebildet werden, weil das beiderseits zu Beginn des Projektes vorhandene Wissen nicht ausreicht. Die Ursachen hierfür liegen in der eingeschränkten Vermittelbarkeit von Erfahrungswissen, der Abstraktheit von Software und dem Wesen moderner Software-Projekte begründet:

a) Der größte Teil der Kenntnisse von Anwendern über ihre konkrete Arbeitssituation ist in einer nicht regelhaft kommunizierbaren Form vorhanden. In oft jahrelanger Zusammenarbeit hat sich zwischen den Mitarbeitern eines Arbeitssystems ein gemeinsames Verständnis über die gemeinsame Arbeit herausgebil-

det, das meist nicht in Form von Regeln oder eindeutigen Vorgaben vorhanden ist. Je komplexer und unstrukturierter Arbeiten sind, desto weniger können die »intuitiv« verfolgten Abfolgen und Zusammenhänge von den Anwendern in einem Arbeitssystem regelhaft verbalisiert werden, selbst wenn sie dies wollten. Mit Blick auf die Entwicklung von Expertensystemen, die sich vorrangig auf den Umgang mit wenig strukturiertem Wissen beschäftigen, verweist Brödner z.B. auf dieses Problem der Akquisition von Wissen: "Es ist nicht böser Wille, der sie (die zu befragenden Experten bei der Wissensakquisition; d.V.) daran hindert, ihr Wissen preiszugeben, sondern der Umstand, daß sie darüber nicht in Form expliziter Regeln verfügen." [Brödner 1985, S. 109]

b) Das Problem, als Experte nicht in formalisierter Form über Wissen zu verfügen, gilt auch für die Entwickler. Auch diese verfügen über ihr Experten- und ihr Erfahrungswissen nicht in Form einfacher Regeln. Schon aus diesem Grund gibt es in Projekten Schwierigkeiten, wenn Entwickler versuchen, Anwendern die Auswirkungen bestimmter technischer Entscheidungen auf den Arbeitsprozeß zu verdeutlichen. Diese Vermittlungsbarriere korrespondiert mit der Tatsache, daß der in Software-Projekten zur Debatte stehende Gegenstand seinerseits äußerst abstrakt[145] ist. Diesem Umstand ist es geschuldet, daß der Wissenstransfer zwischen Entwicklern und Anwendern zusätzlich gehemmt wird. Bei Weltz und Ortmann heißt es mit Blick auf dieses Problem: "Erschwert wurde der Austausch der notwendigen Informationen zwischen Softwareentwicklern und Anwendern durch die Abstraktheit des Gestaltungsgegenstands. Nicht nur war es für die Anwender schwierig, den besonderen Charakter der Anforderungen, die an die Technik gestellt werden, dem Softwareentwickler zu vermitteln, auch umgekehrt fiel es schwer, das Leistungspotential der Technik in vollem Maß anschaulich und nachvollziehbar darzustellen." [Weltz/Ortmann 1992, S. 31]

c) Im Wesen moderner Software-Entwicklungen liegt die dritte Ursache dafür begründet, daß erst gemeinsame Lernprozesse von Entwicklern und Anwendern

---

[145] Weltz und Ortmann führen hinsichtlich der Abstraktheit des Gestaltungsgegenstandes von Software-Projekten aus: "Darin (in der Abstraktheit des Gegenstandes; d.V.) besteht ein wesentlicher Unterschied der Softwareentwicklung zu anderen Entwicklungsprozessen. Das Produkt, sein Leistungsprofil und die gestellte Anforderung sind nur in abstrakten Termini beschreibbar. Diese sind nicht nur meist in einer großen Bandbreite interpretierbar, sie haben unter Umständen für einzelne Beteiligte gar keine oder sogar eine falsche Bedeutung. Natürlich stellt sich dieses Problem in gewisser Weise allen Entwicklungsprozessen in ihren Frühphasen. Anders als im Softwareentwicklungsprozeß allerdings besteht meist die Möglichkeit, relativ rasch diese Abstraktheit- etwa durch Skizzen, Vorentwürfe etc. - zu reduzieren und damit auch den Spielraum des Mißverstehens, unterschiedlicher Interpretationen etc. zu verringern." [Weltz/Ortmann 1992, S. 15]

im Rahmen des Entwicklungsprozesses selbst eine geeignete Grundlage für erfolgreiche Projekte bieten: Solange Technikentwicklung als von der Arbeitsgestaltung isolierter Prozeß bewältigbar schien, wurde auf eine Berücksichtigung der Wechselwirkungen zwischen den neu gestalteten Software-Werkzeugen und dem Arbeitsprozeß verzichtet. Die formalisierbaren Bestandteile der Arbeitsaufgaben wurden identifiziert, in Programme übertragen und in die Arbeitsabläufe re-integriert. Demgegenüber bedeutet Software-Entwicklung heute die Gestaltung eines Arbeitsprozesses. Bei der Entwicklung von Software-Systemen kommt es in diesen Projekten darauf an, ein Arbeitssystem unter Einsatz von Software zu verändern. Das hierfür erforderliche Wissen läßt sich aus mehreren Gründen erst im Laufe des Prozesses selbst als Ergebnis des Austausches von Entwicklern und Anwendern erzeugen. Zunächst ist zu beachten, daß sich das Zielsystem der Anwendung sowie die Rahmenbedingungen des Projektes in beständiger Veränderung befinden. Die Gründe hierfür liegen zunächst in dem sich schnell wandelnden Stand der Technik, der immer wieder neue Möglichkeiten eröffnet, die hinsichtlich der konkreten Erfordernisse zu prüfen sind. Weiterhin liegen sie in den fortlaufenden Veränderungen des Unternehmens und seiner Geschäftspolitik, was zu gravierenden Anforderungsveränderungen an Projekte führen kann. Darüber hinaus spielen die Rückwirkungen erreichter Zwischenstände auf den Entwicklungsprozeß selbst eine Rolle. Je weiter das Gestaltungsproblem durchdrungen wird und je konkreter die gefundenen Lösungen sind, desto besser lassen sich neue Ansatzpunkte und Anforderungen für die Neugestaltung bestimmen. Diese iterative Annäherung an das Gestaltungsoptimum ist bei der Entwicklung neuer Software-Werkzeuge besonders bedeutsam. Werkzeuge lassen sich häufig erst im konkreten Gebrauch auf ihre Einsatztauglichkeit in Arbeitsprozessen prüfen. Entwickler und Anwender lernen in der Auseinandersetzung mit Versionen eines Software-Werkzeugs.

Mittlerweile wird nicht nur in wissenschaftlichen Kreisen akzeptiert, daß Anwender bei der Entwicklung moderner Software-Systeme beteiligt werden müssen.[146]

---

[146] Dies gilt auch für die Entwicklung von Standardsoftware, und zwar in zweifacher Hinsicht. Das Konzept einer Standardsoftware läßt sich heutzutage nur im Hinblick auf die konkrete Anforderungssituation entwickeln. Daher geht der Entwicklung von entsprechenden Systemen immer häufiger der Versuch voraus, die für die jeweilige Anwendungssituation typischen Anforderungen zu identifizieren und Gestaltungsräume für deren jeweilige Spezifik zu lassen. Auf einer anderen Ebene erfordert die Einpassung der Standardsoftware in konkrete Arbeitszusammenhänge eine spezifische Konfiguration des angebotenen Basisprodukts. Oft stellt sich erst in diesem Anpassungsprozeß heraus, daß die angebotene Software nur für ganz spezifische Anwendungssitationen entwickelt wurde und sich der erforderlichen Adaptierbarkeit verschließt. Die Entwicklung des Konzepts der Standardsoftware ist so gesehen ein

Auch in der Praxis lassen sich »Suchbewegungen« nach geeigneten Methoden der Kooperation zwischen Anwendern und Entwicklern beobachten. Hier finden diese Prozesse aber meist als »informelle Initiative« einzelner Projektbeteiligter statt und sind dementsprechend nicht gesondert dokumentiert.

In einer empirischen Untersuchung zu Software-Projekten wird auf die besondere Bedeutung dieser »informellen« Verständigungsversuche[147] von Anwendern und Entwicklern im Projekt hingewiesen. Der Projekterfolg war hochgradig von den nicht »offiziell« angeordneten Aktivitäten bestimmt. Der Erfolg dieser informellen Kooperationsformen liegt offensichtlich in den persönlichen Beziehungen von Entwicklern und Anwendern begründet. Dadurch wurden Vorbehalte und Vorurteile abgebaut, die ansonsten zwischen den beiden Gruppen sehr ausgeprägt vorhanden waren. Interessanterweise bildeten sich hier »naturwüchsig« Kooperationsformen heraus, die faktisch an die Prinzipien modernen Prototypings angelehnt waren. Insgesamt kommen die Autoren zu dem Schluß, daß diese informellen Kontakte sich "als tragfähige und produktive Grundlage für eine weitgehende zeitliche wie inhaltliche Abstimmung der Entwickleraktivitäten an den Bedarf der Anwender" eigneten. [Weltz/Ortmann 1992, S. 75]

Sowohl die »Suchbewegungen« in der Praxis als auch die Forschungsinitiativen wissenschaftlicher Projekte verdeutlichen, daß moderne Projekte einen Bedarf an geeigneten Kooperationsformen zwischen Entwicklern und Anwendern erzeugen.

5 Lernprozesse und Projektorganisation

Kooperation zwischen Entwicklern und Anwendern läßt sich in der Realität auf sehr unterschiedlichen Niveaus beobachten. Sie rangiert von einem Minimum an

---

doppelter Auseinandersetzungsprozeß mit der Anwendungssituation: zum einen mit ihrer jeweiligen Spezifik, die es zu unterstützen gilt, und zum anderen mit ihrer Abstraktion, die Voraussetzung dafür ist, daß es gelingt, besondere Erfordernisse von allgemeinen zu unterscheiden und die Spezifik der Anforderungen auf die darin enthaltenen Basisfunktionen zu beziehen. Für die Gestaltung von Standardsoftware nach partizipativen Prinzipien siehe auch Rauterberg [1993].

[147] Diese informellen Kontakte sind eine Form der sogenannten "stillen Leistungen", die nach Auffassung der Autoren besondere Bedeutung in modernen Projekten spielen. "Stille Leistungen" resultieren aus der erforderlichen Übersetzungsleistung zwischen Anwendern und Entwicklern. Sie sind dann am häufigsten zu beobachten, wenn Entwickler und Anwender bei der Erreichung eines Ziels aufeinander angewiesen sind. Dies ist um so häufiger der Fall, je unstrukturierter die durch den Software-Einsatz zu unterstützende Anwendungssituation ist. Da diese Projekte immer stärker den Hauptteil kommerzieller Projekte ausmachen, schlußfolgern die Autoren, daß die Bedeutung "stiller Leistungen" in der historischen Betrachtung beständig zugenommen habe. [Weltz/Ortmann 1992, S. 112ff.]

Abstimmung zu Projektbeginn bis zu entwickelten Formen der kontinuierlichen Zusammenarbeit über den gesamten Projektverlauf hinweg. Diese Kooperationsformen sind häufig formell festgelegt, so daß ein oder mehrere Vertreter aus den betroffenen Anwendungsabteilungen als Projektteilnehmer benannt werden. Es finden sich in der Praxis allerdings auch Formen der Kooperation, die informell zustandekommen und meist auf der Basis persönlicher Beziehungen zwischen Anwendern und Entwicklern beruhen. [Weltz/Ortmann 1992, S. 72ff.]

Von Heilmann stammt ein Schema, daß die Kooperation zwischen Anwender und Entwickler unter dem **Aspekt der Beteiligung der Benutzer** klassifiziert. Dieses stellt die Reichweite der Benutzerbeteiligung hinsichtlich der Aspekte

- Partizipationsausprägung (keine Partizipation, passive Mitwirkung, aktive Mitentscheidung, Mitwirkung und Mitentscheidung, autonomes Design),
- Partizipationsform (direkte Partizipation, indirekte oder repräsentative Partizipation),
- Partizipationsebene (Arbeitsplatz, Abteilung, Betriebs- und Unternehmensebene),
- Partizipationsgrundlage (formale Grundlage, informeller Konsens),
- Partizipationsphase (in welchen Phase kooperiert wird)[148] dar. [Heilmann 1981, S. 124ff.]

Die unterschiedlichen Formen der Kooperation lassen sich insbesondere hinsichtlich der »Einflußkompetenz« der Anwender unterscheiden. So differenziert Oppermann zwischen drei Stufen der Beteiligung:

- die Informationsstufe beinhaltet Möglichkeiten der passiven, interaktiven oder aktiven Informierung über Betroffeneninteressen,
- die Beteiligung ohne Mitbestimmungsmöglichkeiten enthält stellvertretende, repräsentative und direkte Mitwirkungsformen mit Beratungscharakter,
- die Beteiligung mit Mitbestimmungsmöglichkeiten umfaßt repräsentative oder direkte Beteiligung mit Entscheidungskompetenz. [vgl. Oppermann 1983, S. 34ff.]

---

[148] Diese Unterscheidung ist nur für Projekte sinnvoll, die auf der Basis eines »Wasserfallmodells« organisiert sind. Für moderne Projekte, die aus noch aufzuzeigenden Gründen eher als zyklisches Vorgehen organisiert werden sollten, erscheint diese Unterscheidungsebene wenig hilfreich. Hier kommt es vielmehr darauf an, sich darüber zu verständigen, ob die Anwender kontinuierlich, punktuell oder zeitweilig zu beteiligen sind. [vgl. Mehl u.a. 1993, S. 15]

Die empirische Untersuchung von Weltz und Ortmann verdeutlicht, daß Formen der Beteiligung mit Entscheidungskompetenz in der Praxis nicht die Regel sind. Dort, wo Anwender oder Anwendervertreter beteiligt sind, wird dies oft als Alibi-Funktion gehandhabt. [Weltz/Ortmann 1992, S. 72ff.] Damit sind die notwendigen gemeinsamen Lernprozesse zwischen Anwendern und Entwicklern strukturell blockiert. In den Projekten, in denen Entwickler und Anwender keine stabile Kooperationsbeziehung eingehen, sind lediglich individuelle Lernprozesse einzelner möglich. Gemeinsame Lernprozesse, wie sie für den Erfolg moderner Software-Projekte meist erforderlich sind, sind demgegenüber nur in solchen Projekten möglich, in denen beide Seiten die Gelegenheit haben, eine stabile soziale Beziehung zueinander herzustellen.[149] Die gleichberechtigte Zusammenarbeit von Anwendern und Entwicklern kann als zentrale Voraussetzung einer stabilen Kooperationsbeziehung gelten. Demgegenüber blockieren Kooperationsformen, die lediglich auf eine Beteiligung ohne Mitbestimmungsmöglichkeiten setzen, die Entwicklung einer lerngerechten Kooperationsbeziehung grundsätzlich.

Der Erfolg der Kooperationsbeziehung in Software-Projekten hängt weiterhin von den Methoden ab, die für die Entwicklung einer gemeinsamen Verständigung gewählt werden. In öffentlich geförderten wissenschaftlichen Projekten wird nach Verfahren gesucht, die die erforderlichen Kommunikations- und Lernprozesse zwischen Entwicklern und Anwendern in Rahmen von Software-Projekten unterstützen.[150] Dabei stehen immer häufiger Verfahren im Vordergrund, die primär an der Förderung der sozialen Dynamik in den Projekten ansetzen und die eingesetzten Methoden hinsichtlich der Wirksamkeit für die erforderlichen individuellen und kollektiven Lernprozesse auswählen. [vgl. Daniel 1993]

6 Kooperation im Spannungsfeld betrieblicher Interessen

Das Verhältnis der Kooperationspartner innerhalb eines Software-Projekts ist in der Praxis von unterschiedlichen Vorstellungen der Akteure bestimmt und in ein **Geflecht unterschiedlicher Interessen** eingebettet. Die Bewältigung dieser

---

[149] Aus diesem Grund weisen verschiedene Autoren auf die zentrale Bedeutung der formalen Verankerung der Beteiligungsmöglichkeiten der Anwender hin. Die formale Verankerung der Beteiligungsmöglichkeiten im Rahmen eines Projektvertrags wird besonders in den Arbeiten von Floyd u.a. (STEPS und PEtS) hervorgehoben. [vgl. auch Reisin 1991]

[150] Der Stand der Wissenschaft wird von den hier forschenden Wissenschaftlern allerdings noch sehr zurückhaltend eingeschätzt. Allenthalben wird das Fehlen geeigneter Methoden zur kooperativen Systementwicklung beklagt. [vgl. Coy u.a. 1993]

betriebspolitischen Anforderung ist ein zentraler Erfolgsfaktor von Software-Projekten. [Weltz/Ortmann 1992, S. 128]

Entwickler und Anwender trennen häufig sehr unterschiedliche »Weltbilder« und damit verbundene Wahrnehmungen der betrieblichen Realität. Erschwert wird die Beziehung beider Seiten darüber hinaus häufig durch Vorurteile und falsche Vorstellungen über die jeweils andere Gruppe.[151] Entwickler unterstellen Anwendern oft die Unfähigkeit, Anforderungen präzise zu formulieren und die Arbeit des Entwicklers durch immer neue Wünsche zu erschweren. Die Anwender ihrerseits weisen darauf hin, daß die Entwickler keine Vorstellung von der Komplexität der Arbeitswirklichkeit und wenig Verständnis für die Besonderheit der Situation hätten. Sie meinen, daß Entwickler nur auf die Technik fixiert wären und vom wirklichen Leben abgeschottet in einer DV-Abteilung oder einem Softwareunternehmen »vor sich hinwerkeln« würden. Letztlich resultierte aus diesen Problemen, so die Untersuchung von Weltz und Ortmann, daß zwar prinzipiell die Notwendigkeit der Kooperation in Software-Projekten postuliert, in der Praxis aber weitgehend aneinander vorbei gearbeitet wurde. So schreiben z.B. die Autoren: "Viele Softwareentwickler zogen daraus die Folgerung, daß »Beteiligung« nichts bringe, daß man die Kontakte mit den Nutzern möglichst gering halten müsse und den Nutzer als Informationsquelle, nicht aber als Gesprächspartner behandeln müsse." [ebd., S. 77]

Diese unterschiedlichen Wahrnehmungen sind aber keineswegs nur Ergebnis »menschlicher Schwächen«. Meist stehen hinter diesen unterschiedlichen Wahrnehmungen und Vorstellungen sehr konkrete Ursachen. Das Konfliktpotential zwischen Entwicklern und Anwendern - aber auch unter Anwendern selbst - resultiert meist aus sehr unterschiedlichen Zielvorstellungen und Wünschen hinsichtlich der gemeinsamen Gestaltungsaufgabe, die sich oft aus unterschiedlichen Interessen speisen.

- So wurde in der oben genannten empirischen Untersuchung beobachtet, daß das Verhältnis von Entwicklern und Anwendern durch sehr unterschiedliche Grundorientierungen bei der Interpretation der Projektaufgabe bestimmt war. Entwickler stellen häufig technikimmanente Aspekte der Gestaltungsaufgabe in den Vordergrund. Die von ihnen angestrebte Qualität des Produkts bezog sich vorrangig auf technische Qualitätskriterien. Demgegenüber steht für Anwender vorrangig der Aufgaben- und Anwendungs-

[151] Interessant ist in diesem Zusammenhang der Hinweis von Weltz und Ortmann, daß die Vorurteile zwischen Entwicklern und Anwendern umso deutlicher anzutreffen waren, je formalisierter und sporadischer der Kontakt zwischen den beiden Gruppen war. [vgl. Weltz/Ortmann 1992, S. 76]

bezug während des Projekts im Vordergrund. Ihre Ziele waren daher eher »Erleichterung bei der Aufgabenerledigung« oder »Belastungsreduzierung«. [vgl. Heilmann 1981]

- Die Differenzen zwischen verschiedenen Anwendern resultierten häufig daraus, daß mit der Gestaltung des Arbeitsprozesses die Arbeitsaufgaben oder das Verhältnis zwischen verschiedenen Abteilungen verändert werden sollte. Gerade diese Interessenauseinandersetzungen werden oft hinter technischen Anforderungen versteckt.
- Differenzen innerhalb der Entwicklergruppe ergaben sich daraus, daß mit bestimmten technischen Entscheidungen konkrete Vor- oder Nachteile verbunden waren. Mit der Wahl einer bestimmten Programmiersprache wurden z.B. die Karriereaussichten und Qualifizierungschancen bestimmter Entwicklergruppen nachhaltig beeinflußt. [Weltz/Ortmann 1992, S. 125ff.]

Die Differenzen und Interessengegensätze sind ursächlich darauf zurückzuführen, daß Software-Projekte meist tief in das betriebliche Gleichgewicht von Einfluß und Kontrolle eingreifen und die Arbeitsbedingungen sehr grundlegend verändern. Weil die Entwicklung von Software auf die Gestaltung eines Arbeitsprozesses zielt, ist die betriebspolitische Verarbeitung der damit einhergehenden Konflikte untrennbar mit dem Fortgang des Projekts verbunden. Software-Projekte können daher nicht jenseits dieser Auseinandersetzungen zum Erfolg gebracht werden, sondern nur durch einen produktiven Umgang mit ihnen. Hierin liegt eine zentrale Anforderung an moderne Software-Projekte.[152]

7 Lernen und Konsens im Prozeß

Moderne Software-Projekte müssen so organisiert sein, daß sie einen gemeinsamen Lernprozeß zwischen den Projektbeteiligten, insbesondere zwischen Anwendern und Entwicklern fördern. Dies setzt voraus, daß die **Dynamik der sozialen Prozesse** unterstützt wird, daß also Konflikte und Interessenauseinandersetzungen in einem offenen und argumentativen Diskurs ausgetragen werden. Diese Anforderungen erfüllt das klassische Phasenmodell, das zumindest der Theorie nach die Grundlage heutiger Software-Entwicklung ist, nicht.[153] Das

---

[152] Sehr anschaulich werden die politischen Prozesse zur Gestaltung von Arbeit und Technik von Ortmann u.a. [1990] dargestellt. Gutes Anschauungsmaterial findet sich auch in Weltz/Ortmann [1992].

[153] Phasen- oder Wasserfallmodelle werden in der Praxis offensichtlich deshalb angewandt, weil sie die Illusion erzeugen, die Komplexität moderner Software-Entwicklung sei schematisch planbar. [vgl. Weltz/Ortmann 1992] In unseren Gesprächen sind wir darüber hinaus auf zwei weitere Gründe gestoßen, die die Verwendung von

klassische Phasenmodell, welches die schematische, sequentielle Abfolge einzelner Phasen wie Problemanalyse, funktionale Analyse, Entwurf, Programmierung, Funktions- und Leistungsüberprüfung, Installation und Wartung vorsieht, baut nach Hesse auf folgenden Prämissen auf:

- alle für die Entwicklung der Software erforderlichen Vorgaben sind zu Beginn des Projektes vollständig definierbar,
- das hierfür erforderliche Wissen ist zu Beginn des Projektes vorhanden,
- alle Vorgaben des Projektes werden konsensual von allen getragen. [Hesse 1991, S. 18f.; zit.n. Weltz/Ortmann 1992, S. 138]

Der Realität moderner Software-Projekte entsprechen diese Prämissen aber nicht. Das erforderliche Wissen entsteht erst nach und nach im gemeinsamen Lernprozeß von Anwendern und Entwicklern. Die Vorgaben lassen sich keineswegs am Beginn des Projektes, sondern nur Schritt für Schritt definieren. Ähnliches gilt für den Prozeß der Konsensfindung. Er verläuft als vielfach gebrochener Prozeß, in dem anfangs scheinbar bestehende Übereinstimmungen mit zunehmendem Wissen häufig ihre Gültigkeit verlieren und immer wieder neu bestimmt werden müssen. Deutlicher Ausdruck für die schrittweise und evolutionäre Entwicklungsdynamik von Software-Projekten ist die Tatsache, daß etwa zwei Drittel aller Softwarekosten auf Nachbesserung und Änderungen vorhandener Programme entfallen.[154] [IDC 1989; zit.n. Weltz/Ortmann 1992] Das Phasenmo-

---

Phasenmodellen in der Praxis sinnvoll erscheinen lassen. Einerseits werden diese Projektmodelle deshalb verwandt, weil sie von den Auftraggebern vorgeschrieben werden. Insbesondere öffentliche Auftraggeber fordern die Planung und Abrechnung von Projekten nach diesem Schema. Die Projekte verfahren in diesen Fällen dann meist so, daß sie in ihrer »Außenpolitik« das Projekt nach dem Phasenmodell steuern, während sie dies in der »Innenpolitik« aus Praktikabilitätserwägungen tunlichst unterlassen. Weiterhin werden Phasenmodelle da verwandt, wo ein verwendetes CASE-Tool auf diesem Modell basiert. Darauf verweisen auch Bittner/Hesse [1993] anhand einer empirischen Untersuchung zur Verwendung von Methoden in Software-Projekten.

[154] Diese auf den ersten Blick verblüffende Angabe ist nicht nur darauf zurückzuführen, daß die Wartung von Software sehr viel Arbeitszeit von Informatikfachkräften in Anspruch nimmt. Der hohe Anteil der Kosten für Änderungen und Nachbesserungen kommt auch durch die schematische Anwendung von Phasenmodellen bei der Projektabwicklung zustande. In unseren Expertengesprächen wurde uns berichtet, daß nach dem »Wasserfallmodell« abzuwickelnde Projekte den größten Teil ihres Budgets bereits aufgebraucht haben, wenn die eigentlichen Anpassungen an die Anwendungsrealität - angestoßen durch die Auseinandersetzung mit den Kunden bei der Implementierung der Software - beginnen. Diese nachträglichen Anpassungen, die eigentlich den Charakter originärer Entwicklungen haben, führen zu einer Aufblähung der Kosten für die Software-Entwicklung in der Wartungsphase, obwohl der Projektplan dies nicht vorsieht. So bricht sich dann die Realität Bahn, daß moderne Software-Systeme nur in kontinuierlicher Auseinandersetzung mit der konkreten Situation

dell, so Weltz und Ortmann, behindert angesichts dieser Situation sogar den Projekterfolg: "Insgesamt war erkennbar, daß die Vorstellung, Entwicklungsprozesse ließen sich mit Hilfe des Phasenschemas stringent planen und steuern, sich eher als Hypothek erwies und vielfach teuer erkauft werden mußte." [Weltz/Ortmann 1992, S. 142]

Hinsichtlich der Organisation der Software-Projekte besteht die Anforderung, die Prozesse des gemeinsamen Lernens und der Konsensbildung mit der Gestaltung des Arbeitsprozesses zu verbinden. Software-Entwicklung muß als "ongoing social and political process" [Kling 1987] gestaltet werden.[155] Daraus resultiert die Notwendigkeit, Projektverfahren einzuführen, die sich der **evolutionären Bewegungsform** von Lernprozessen und Prozessen der Konsensbildung anpassen. Charakteristisch dafür ist es, die Entwicklung ausgehend von der sozialen Dynamik im Software-Projekt her zu steuern, statt entlang der technischen Entwicklungsstufen der Software-Systeme. Mit Blick auf die erforderliche Verbindung der einzelnen Prozesse fassen die Autoren der Untersuchung zu Software-Projekten ihre Erfahrungen folgendermaßen zusammen: "Erfolgreich hingegen waren Projekte, in denen der technische Entwicklungsprozeß mit Prozessen der Konsensbildung und der Wissensakquisition verzahnt und so eine laufende Neubestimmung sowohl der Entwicklungsziele wie der Projektabwicklung erreicht werden konnte." [Weltz/Ortmann 1992, S. 159]

Diesen zentralen Anforderungen an die Organisation von Software-Projekten folgend werden seit den achtziger Jahren verstärkt Projektkonzepte entwickelt, die sich von herkömmlichen Ansätzen der Software-Entwicklung in erster Linie durch die gezielte Orientierung auf die Evolution von Systemen [Floyd 1987] unterscheiden. Ein solches Verfahren stellt beispielsweise der Projekt- und Methodenansatz STEPS dar. Dieses Verfahren baut auf folgenden Prinzipien auf:

- Einer prozeßorientierten Sichtweise der Softwareentwicklung, statt einer ausschließlichen Orientierung auf das Software-Produkt.
- Einem menschzentrierten statt einem technikzentrierten Qualitätsbegriff.

---

im zu gestaltenden Arbeitssystem entwickelt werden können. Auch wenn alle Beteiligten an einem Projektmodell festhalten, daß gerade dies verhindert soll.

[155] Auch wenn wir uns in den folgenden Ausführungen vornehmlich auf die Lernprozesse der IT-Fachkräfte konzentrieren, darf selbstverständlich nicht außer acht gelassen werden, daß auch auf Seiten der beteiligten Anwender komplementäre Lernprozesse notwendig für das Gelingen der Projekte sind und eine entsprechende Beteiligungskompetenz erforderlich ist. Für den Zusammenhang von Beteiligung und Kompetenz auf Seiten der Anwender siehe Kißler [1990].

- Einem zyklischen Phasenmodell, das auf die Erstellung, Benutzung und Revision von Versionen des Systems gerichtet ist statt auf die linear-sequentielle Erstellung eines einmaligen Produkts.
- Aufeinander abgestimmten Methoden für die Anforderungsermittlung, die Dialoggestaltung und den Software-Entwurf.
- Dem Einsatz von Techniken, die auf die Entfaltung und Förderung von Kommunikations- und Lernprozessen zwischen Entwicklern und Benutzern zielen. [vgl. Mehl u.a. 1993, S. 3]

8 Zusammenfassung

Software wird in der Praxis im Rahmen von Software-Projekten entwickelt. Diese, durch die zeitliche Befristung sowie die personelle und inhaltliche Abgegrenztheit definierte Organisationsform vereint Experten aus verschiedenen Fachgebieten und Unternehmensbereichen unter einer gemeinsamen Zielstellung. Besonderes Charakteristikum von Projekten ist deren Einmaligkeit. Kommerzielle Software-Projekte haben meist die Entwicklung sozio-technisch eingebetteter Software zum Gegenstand. In diesen Projekten ist die Gestaltung eines Arbeitssystems mit Hilfe softwaretechnischer Systeme das Ziel. Die Qualität in diesen Projekten bestimmt sich in erster Linie aus der Verwendbarkeit der Software im Arbeitszusammenhang. Das stellt besondere Anforderungen an die Entwickler und den Software-Entwicklungsprozeß.

Das erforderliche Wissen zur Entwicklung moderner Software-Systeme läßt sich nur in gemeinsamen Lernprozessen von Entwicklern und Anwendern nach und nach entwickeln. Dabei bringen beide Gruppen jeweils komplementäres Vorwissen in den gemeinsamen Kooperationsprozeß ein. Die Entfaltung gemeinsamer Lernprozesse zwischen Anwendern und Entwicklern setzt eine stabile soziale Beziehung zwischen den beiden Gruppen voraus. Diese basiert auf einer gleichberechtigten Kooperationsbeziehung mit entsprechenden Entscheidungskompetenzen der beiden Gruppen. Förderlich sind solche Kooperationsmethoden, die die Dynamik der sozialen Prozesse sowie die individuellen und kollektiven Lernprozesse im Projekt unterstützen.

Konflikte und Interessenauseinandersetzungen sind zwingender Bestandteil moderner Software-Projekte. Sie zu umgehen und sich den vermeintlich leichter zu lösenden technischen Fragestellungen zuzuwenden, führt meist zu einer Verschleppung der Konflikte und ist Ursache vieler Projekt-Krisen. Projekterfolg ist unter diesen Bedingungen nur möglich, wenn es gelingt, die vorhandenen Konflikte und Interessengegensätze offen auszutragen. Klassische Phasenmodelle werden diesen Anforderungen an moderne Software-Entwicklung in der

Praxis nicht mehr gerecht. Benötigt werden Projektkonzepte, die an der evolutionären Entwicklungsform des Gestaltungsprozesses orientiert sind und die einzelnen Teilprozesse des gemeinsamen Lernens und der Konfliktbewältigung produktiv aufeinander beziehen.

III Software-Entwicklung und Arbeitsgestaltung

Im Zusammenhang mit der Entwicklung und dem Einsatz moderner Informations- und Kommunikationssystemen sind die Einflüsse dieser Technologie auf die Gestaltung von Arbeitsplätzen und der Aufbau- und Ablauforganisation von Arbeit zu einem vieldiskutierten Thema geworden. Einige Informatiker sehen Konsequenzen für die Theorie der Informatik, die sich bisher allein auf formalisierte Strukturen konzentriert hat und nun zunehmend den Einsatzkontext der Informatik berücksichtigen muß. Unternehmen stellen die strategische Bedeutung der IT fest und fordern die Integration der IT-Entwicklung in die betrieblichen Innovationsprozesse. Im Bereich des Software Engineering werden Möglichkeiten diskutiert, Software-Entwicklungsprozesse an den Anforderungen der Arbeitsgestaltung auszurichten.

Das Software-Projekt steht mit seinen Aufgaben mitten im Geschehen betrieblicher Innovationsprozesse. Hier muß es Informatikern gelingen, die unternehmensspezifischen Anforderungen an die Gestaltung der Arbeit umzusetzen. Unabhängig davon, ob das Software-Projekt mit dem expliziten Auftrag der Neugestaltung der betroffenen Arbeitsabläufe betraut ist, oder ob diese Anforderung hinter einem technisch formulierten Projektauftrag verborgen ist: Mit der Entwicklung von Software, die der Unterstützung von Arbeit dient, wird auch Arbeit gestaltet.

- Die Verteilung der Funktionen zwischen Mensch, Computer und anderen Arbeitsmitteln wird neu bestimmt.
- Die Arbeitsteilung zwischen den Menschen wird verändert.
- Die Kooperation der Mitarbeiter des Unternehmens muß an die Möglichkeiten und Grenzen der neuen Software angepaßt werden. Neuer Bedarf an kommunikativer Abstimmung entsteht und neue Regeln, die das gemeinsame Arbeiten koordinieren, müssen eingeführt werden.
- Arbeitsverfahren müssen an die neuen Bedingungen angepaßt werden.
- Die notwendigen Qualifikationen zur Bewältigung der Arbeit haben sich mit der Software verändert.
- Unter Umständen ist die Regelung der Arbeitszeit vom Einsatz der Software betroffen.
- Der Einsatz der Software hat Auswirkungen auf die physischen und psychischen Beanspruchungen am Arbeitsplatz.

Für das Software-Projekt steht also nicht die Frage des »Ob«, sondern allein die Frage des »Wie«.

In einem ganzheitlichen Verständnis ihrer beruflichen Aufgaben müssen IT-Fachkräfte diesen Zusammenhang von Software-Entwicklung und Arbeitsgestaltung realisieren und die notwendige Handlungskompetenz entwickeln. Die Qualifizierung muß hier einen geeigneten Bezugsrahmen bieten. Sie muß unter der Bearbeitung des notwendigen Wissens, der sozialen Kompetenzen und der beruflichen Identität in der Software-Entwicklung dazu beitragen, daß eine auf Arbeitsgestaltung ausgerichtete Handlungskompetenz aufgebaut werden kann.

Das Kapitel III »Software-Entwicklung und Arbeitsgestaltung« beschreibt die inhaltlichen Grundlagen zur Vermittlung einer Technikgestaltungskompetenz, die systematisch in den Kontext der Arbeitsgestaltung eingebettet ist.

1 Arbeitsgestaltung im Software-Projekt

1.1 Leitbilder der Software-Entwicklung

Mit der Entwicklung von Software ist die Frage nach der Funktionsteilung zwischen Mensch und Computer sowie der optimalen Einbettung des Computers in den Arbeitsprozeß verbunden. Bestimmte Arbeitsanteile werden aus dem Arbeitsprozeß herausgelöst, auf den Computer übertragen und dann in den Arbeitsprozeß reintegriert.

Um Software entwickeln zu können, müssen IT-Fachkräfte davon ausgehen, daß Arbeit zumindest in Teilen formalisierbar und algorithmisierbar ist. Ohne diese Annahme ist die Entwicklung von **Anwendungssoftware in einem soziotechnischen Kontext** nicht möglich. Dem steht allerdings entgegen, daß in der Arbeitswelt eindeutige mathematische Operationen nur eine geringe Rolle spielen. Statt dessen ist das Software-Projekt mit situations- und personenabhängigen Handlungen und Tätigkeiten konfrontiert, für die die Annahme der Formalisierbarkeit nur sehr eingeschränkt gilt. **Formalisierung von Arbeit** ist immer mit modellhaften Vorstellungen von Arbeit verbunden, die von den meisten Gegebenheiten des Arbeitsalltags abstrahieren. Eine solche vereinfachende Vorstellung liegt beispielsweise den Methoden der Funktionsanalyse wie etwa der Structured Analysis [vgl. z.B. Raasch 1991] zugrunde. Arbeit wird auf Funktionen und Daten reduziert, mit deren scheinbar objektivierbaren Regeln Situations- und Personenabhängigkeit des Handelns per definitionem ausgeschlossen wird.

Um Software entwickeln zu können, benötigen IT-Fachkräfte derartige Modelle, für die in einigen Bereichen von der Vielfältigkeit konkreter Handlungssituationen

abstrahiert und auf eine endliche Zahl von Handlungsmöglichkeiten reduziert wird. Dieser Abstraktionsschritt ist allerdings mit Nachteilen verbunden, die jeder Entwickler und jedes Software-Projekt abwägen muß: Das Modell von Arbeit wirkt über den Einsatz der Software auf die Arbeit zurück und zwingt den Menschen, sich auf die darin enthaltene Formalisierung der Arbeit einzustellen. Mit Software werden dann Handlungsmöglichkeiten und -abläufe sowie Entscheidungswege vorgegeben, die die Fähigkeit flexibler Reaktionen auf Veränderungen und die Möglichkeit der Herausbildung individueller Arbeitsstile stark einschränkt.

Die Algorithmisierung von Arbeit besteht im wesentlichen aus diesen Abstraktionsschritten, die etwa bei der Abbildung technischer Prozesse eine nur geringe Rolle spielen. Sie ist kein rein methodisch zu lösendes Problem, sondern ein Entscheidungsproblem, das nur im Rahmen der Arbeitsgestaltung gelöst werden kann.

Die Formalisierung von Arbeitsanteilen orientiert sich heute an zwei Leitbildern, denen sehr unterschiedliche Ansätze der Arbeitsgestaltung zugrunde liegen. Wir bezeichnen diese Leitbilder als »Abbilden« und »Unterstützen«.

Das **Leitbild des »Abbildens«** in der Software-Entwicklung orientiert auf eine Übertragung (=Abbildung) ganzer Handlungsabläufe auf den Computer. Algorithmisierbare Operationen werden über stark formalisierte Entscheidungsmuster (etwa in Form von Wenn-Dann-Beziehungen mit und ohne Eingriff des Anwenders) miteinander verknüpft. Software-Entwicklung, die am Leitbild des »Abbildens« orientiert ist, führt oft zu sehr großen Software-Systemen, die über relativ wenige Schnittstellen in die verbliebene lebendige Arbeit integriert werden. Sie sind durch die Dateneingabe, die Datenausgabe und eine endliche Anzahl manipulierbarer Bearbeitungszustände zu kennzeichnen.

Das Leitbild des »Abbildens« korrespondiert mit Ansätzen der Arbeitsgestaltung, die eine Formalisierung der gesamten Arbeit anstreben. So basieren viele Unternehmensstrategien bisher auf dem tayloristischen Ansatz, der zu einer tiefen Arbeitsteilung und einer sehr weitgehenden Strukturierung von Arbeitsabläufen führt. Für die Realisierung quantitativer Unternehmensziele war die Einengung von Handlungsspielräumen, die routinemäßige Aneinanderkettung von Handlungen und die Formalisierung von Entscheidungen erwünscht.

Demgegenüber orientiert das **Leitbild des »Unterstützens«** auf einen eher punktuellen Software-Einsatz und die möglichst geringe Einengung von Handlungsmöglichkeiten. Da auf die Abbildung von Entscheidungsmustern weitgehend verzichtet wird, sind die in Software abgebildeten algorithmisierbaren Operationen nicht fest verknüpft, sondern flexibel einsetzbar.

Nach diesem Ansatz entstehen kleine Objekte und Module, die in nur losem, nicht fest definierten Verhältnis zueinander stehen. Da prinzipiell eine beliebige Anzahl von Systemzuständen und Handlungsabläufen mit diesen Objekten realisierbar ist, ist das Verhältnis der lebendigen Arbeit zur Software nicht mehr über eine relativ geringe Anzahl von Schnittstellen zu erfassen. Die lebendige Arbeit und die eingesetzten Software-Objekte stehen in einem komplexen, vielschichtigen Verhältnis zueinander, in dem Software-Objekte vom Menschen situationsabhängig untereinander verknüpft werden.

Das Leitbild des »Unterstützens« korrespondiert mit modernen Ansätzen der Arbeitsgestaltung, die qualitative Unternehmensziele über eine Flexibilisierung und Qualitätsorientierung der Arbeitsprozesse realisieren wollen. Die Vielfalt von Handlungsmöglichkeiten und Entscheidungssituationen soll hier erhalten und effektiv durch Software unterstützt werden. Die Vereinfachung und Abbildung von Handlungsabläufen ist nicht erwünscht.

Die Leitbilder »Unterstützen« und »Abbilden« sind an jeweils spezifischen Merkmalen der Arbeit orientiert. In der Formalisierung und Abbildung ganzer Handlungsabläufe genügt eine Orientierung an der äußeren Struktur, an Daten, Funktionen sowie zeitlichen oder logischen Abläufen. Ihre Komplexität wird soweit reduziert, bis sie programmierbar sind. Für die Unterstützung von Arbeitsabläufen bietet diese Perspektive nur wenige Ansatzpunkte. Im Gegenteil muß davon ausgegangen werden, daß sie bereits modellhafte Vorstellungen von Arbeit impliziert, die eine umfassende Abbildbarkeit suggerieren. Die ausschließliche Orientierung an diesen äußeren Strukturmerkmalen der Arbeit stellt sich so als Behinderung der Entwicklung unterstützender Software-Werkzeuge heraus.

Für die Unterstützung von Arbeitsabläufen benötigt das Software-Projekt also eine andere Perspektive, die der Komplexität von Arbeit Rechnung trägt und doch Ansatzpunkte einer sinnvollen Formalisierung bietet. Auf der Ebene der Arbeit des einzelnen Menschen bietet die Orientierung an der **Handlungsregulation** diese Ansatzpunkte. Ein Verständnis von Arbeit als Einheit zielsetzender, vorbereitender, ausführender und kontrollierender Tätigkeiten unter bestimmten Bedingungen der Handlungsregulation ist die Grundlage der Gestaltung von Software, die Arbeitsabläufe nicht abbildet, sondern effektiv unterstützt. Auf der Ebene kooperativer Arbeit bietet die Orientierung an der **sozialen Regulation** diese Ansatzpunkte. Zeitliche, räumliche und sachliche Verhältnisse von Tätigkeiten begründen Kooperationserfordernisse, die insbesondere durch die Unterstützung der Kommunikation zum Gegenstand der Software-Entwicklung werden.

Moderne Software-Projekte sind mit den Möglichkeiten, die die Informationstechnik bietet, und den Konzepten der Arbeitsgestaltung, die heute in vielen Anwenderunternehmen verfolgt werden, zunehmend dem Leitbild des »Unterstützens« verpflichtet. Im Gegensatz zur Software-Entwicklung als Abbildung, die eine Externalisierung der Arbeitsgestaltung erlaubt, muß die Entwicklung unterstützender Software in einen Prozeß der Arbeitsgestaltung eingebettet werden. Mangels definierbarer Schnittstellen muß die Optimierung von Arbeitsabläufen im direkten Verhältnis abgebildeter Operationen, menschlicher Handlungsregulation und sozialer Koordination erfolgen.

1.2 Ziele der Arbeitsgestaltung im Software-Projekt

Mit dem Software-Projekt werden vom Auftraggeber und von den Anwendern Ziele verfolgt, die sich nur selten in der technischen Qualität eines neuen Software-Produkts erschöpfen. Indem Software-Projekte Arbeit gestalten, sind sie Zielen wie

- der Effektivität der Arbeit,
- der Optimierung der Beanspruchung der Fähigkeiten der Mitarbeiter und der technischen Optionen,
- der Gesundheitsförderlichkeit und
- im Zuge neuer Arbeitseinsatzkonzepte auch der Persönlichkeitsförderlichkeit

verpflichtet. [vgl. Hacker 1987, S. 35]

Das **Gestaltungspotential**, das dem Software-Projekt zur Verwirklichung dieser Ziele zur Verfügung steht, reicht von der Realisierung der Benutzungsfreundlichkeit von Software bis hin zur aktiven Gestaltung von Arbeit.

Der Ansatz der **Benutzungsfreundlichkeit** geht davon aus, daß Behinderungen und Beeinträchtigungen durch Software minimiert werden sollen. Die in der Funktionsteilung von Mensch und Computer gegebenen Schnittstellen sollen dem Menschen und seiner Arbeitsweise optimal etwa durch ergonomische Bildschirmgestaltung und Hilfefunktionen angepaßt werden. Da dieser Ansatz die Funktionsteilung zwischen Mensch und Computer weitgehend unbeachtet läßt, wird er durchaus auch kritisch betrachtet. Er führt zu einer Verabsolutierung von Kriterien, die je nach Arbeitsplatzbedingungen differenziert zu betrachten sind, wie etwa die einfache Bedienbarkeit von Software: "Eine einfache (=anforderungsarme) Benutzungsschnittstelle kann sich nur mit anspruchsvollen (anforderungsreichen) Tätigkeitsinhalten vertragen. Ansonsten (etwa bei repetitiven Teilarbeiten) käme es unweigerlich zu einer Unterforderung in der Arbeits-

tätigkeit mit den bekannten negativen Auswirkungen (wie z.B. Monotonie mit demotivierenden Langzeitfolgen)." [vgl. Paetau 1990, S. 176]

Der Ansatz wird auf verallgemeinerbare Gestaltungskriterien reduziert, die es angesichts der Vielfalt von Arbeitsbedingungen und -weisen nicht geben kann. Die äußere Form von Software kann für den einen Arbeitsprozeß positive, für den anderen Arbeitsprozeß negative Auswirkungen haben, so daß die konkrete Entscheidung allein im Kontext eines umfassenden Gestaltungsansatzes getroffen werden kann. [vgl. Hacker 1987, S. 34]

Die Anpassung von Schnittstellen eröffnet für das Software-Projekt nur geringe Gestaltungspotentiale. Benutzungsfreundlichkeit muß statt dieser verengten Sicht in ein **Gesamtkonzept der Gestaltung menschlicher Arbeit** eingebettet sein, das

- die Ebene des Arbeitsplatzes als dem Tätigkeitsbereich einer Person und
- die Organisation, in die die Kooperation und die sozialen Verhältnisse arbeitender Menschen eingebettet sind,

umfaßt.

Erst wenn die Frage nach der Form der Unterstützung der Arbeit durch die Software geklärt ist, kann die Frage nach einer sinnvollen Gestaltung der Benutzungsschnittstelle sinnvoll beantwortet werden.

Hacker verknüpft die Gestaltung von Arbeit mit Gestaltungszielen, die für die gesamte arbeitswissenschaftliche Diskussion in diesem Bereich grundlegenden Charakter haben: **Ausführbarkeit, Schädigungslosigkeit, Beeinträchtigungslosigkeit und Persönlichkeitsförderlichkeit** von Tätigkeiten. [ebd., S. 34ff.]

Die *Ausführbarkeit* wird im allgemeinen als so selbstverständlich angesehen, daß sie kaum noch zur Sprache kommt und in ihren Dimensionen unterschätzt wird. Die »normale« Arbeitssituation geht in ihrer Form der Belastung und der Dauer weit über eine konstruierte, meist kurzfristige Testsituation unter weitgehend beeinträchtigungsfreien Laborbedingungen hinaus. Ausführbarkeit wird im Arbeitsalltag zu einer kognitiven Ausführbarkeit in spezifischen Belastungssituationen, die durch das Überfordern menschlicher Informationsverarbeitungsvorgänge in Frage gestellt wird. Hacker identifiziert drei Typen von Mängeln, die trotz vollständiger Funktionsfähigkeit der Software zu einer Nicht-Ausführbarkeit von Tätigkeiten führen:

- Eine psychisch wirksame Information über ein erforderliches Handeln fehlt. Nicht jede technisch realisierte Information ist auch psychisch wirksam. Dies kann etwa auf Zeitverzögerungen zwischen Ereignis und Information beru-

hen oder auf einer geringen Trennschärfe von Informationen, etwa bei Tönen oder schriftlichen Meldungen.

- Eine Information ist zwar psychisch wirksam, kann aber in der Handlungsregulation nicht genutzt werden. Dieser Fall ist etwa dann gegeben, wenn die Zeitspanne zwischen Informationserhalt und der Möglichkeit der Umsetzung zu lang ist, die Information also vergessen wurde.
- Informationen verführen zu Fehlinterpretationen. Die spontane Interpretation von Informationen ist abhängig von Interpretationsgewohnheiten, die gesellschaftlich etabliert sein können, sich in einem Berufsfeld entwickelt haben oder auch nur den Gewohnheiten einer Abteilung entsprechen. Informationen, die sich an diese Kodierung nicht halten, führen leicht zu einer falschen Nutzung. Hacker führt etwa das Beispiel an, Kühlwasser mit der Farbe Rot und heißes Wasser mit der Farbe Blau zu belegen. [ebd., S. 35]

In modernen Arbeitsprozessen bekommen diese Aspekte von Ausführbarkeit eine zunehmende Bedeutung. Paetau weist darauf hin, daß sich durch die Einführung neuer Techniken das Tempo von Produktionsprozessen beschleunigt: "Vor allem durch den Einsatz informationstechnisch gesteuerter Maschinen hat sich die Präzision in der Arbeit, der Umfang und die Geschwindigkeit der Informationsaufnahme und -verarbeitung beträchtlich erhöht. (...) Forschungsergebnisse zeigen, daß in vielen Fällen die psychologischen Reaktionsfähigkeiten des menschlichen Organismus überfordert sind." [Paetau 1990, S. 178] Tritt eine Überforderung ein, wird die Komplexität von Tätigkeiten unkontrolliert reduziert und die angestrebte Effektivität nicht erreicht.

Das Kriterium der *Schädigungslosigkeit* impliziert den Ausschluß arbeitsbedingter körperlicher, psychonervaler und sozialer Schädigungen.

Der Einsatz von Computern führt zu einer Zunahme psychischer Belastungen, die zu einer erheblichen Beeinträchtigung des Befindens und damit zu Effektivitätsminderungen im Arbeitsprozeß führen können. Das Kriterium der *Beeinträchtigungslosigkeit* umfaßt deshalb den Ausschluß überdurchschnittlicher Ermüdungs-, Monotonie- und Sättigungsgrade.

In den aktuell diskutierten Konzepten der Arbeitsgestaltung kommt der *Persönlichkeitsförderlichkeit* von Tätigkeiten eine zentrale Bedeutung zu. Die Förderung und Entwicklung der Persönlichkeit und die Möglichkeit zum Hinzulernen in der Arbeit stehen in enger Beziehung zu einer intrinsischen Arbeitsmotivation, die für die Bewältigung komplexer unstrukturierter Arbeitsaufgaben maßgeblich ist.

Diese Kriterien gehören heute zum Standard der Arbeitswissenschaften und dienen verschiedenen Ansätzen der Arbeitsgestaltung als Orientierungspunkt.[156]

Die Umsetzung dieser Gestaltungskriterien im Software-Projekt berührt die Ebene der Arbeitsinhalte und die Ebene der subjektiven Regulation von Tätigkeiten. Die Merkmale von Arbeitstätigkeiten basieren auf den Möglichkeiten und Merkmalen kognitiver Handlungsregulation. Hacker unterscheidet

- eine intellektuelle Regulationsebene durch die gedankliche Vorwegnahme und Zerlegung von Tätigkeiten,
- eine perzeptiv-begriffliche Regulationsebene durch Signale und Sprache
- eine sensumotorische Regulationsebene unterhalb der Bewußtseinsebene. [vgl. Hacker 1986]

Für die Gestaltung von Arbeitstätigkeiten müssen im Software-Projekt die Bedingungen der Handlungsregulation optimiert werden, um die Gestaltungskriterien Ausführbarkeit, Schädigungslosigkeit, Beeinträchtigungslosigkeit und Persönlichkeitsförderlichkeit zu verwirklichen.

Die arbeitswissenschaftliche Diskussion ist sehr stark an den psychologischen Grundlagen der **Handlungsregulation des Individuums** orientiert. Dies ist aber nur eine Ebene der Arbeitsgestaltung. Im Zusammenhang mit dem Einsatz der Informationstechnik rückt zunehmend auch ihre Bedeutung für die Neugestaltung von Organisation und Kooperation ins Blickfeld.

Die Gestaltungsmöglichkeiten moderner Informationstechnik verändern die **Bedingungen der Kooperation** grundlegend. Als zentrale Ursache hebt Paetau besonders das **Integrationspotential** hervor, das moderne Hardware- und Software-Architekturen bieten. [Paetau 1990, S. 196] Es sind zwei Ebenen der Integration von Elementen der Informationsverarbeitung festzustellen: auf der Ebene der Hardware- und Software-Konfiguration und auf der Ebene der Informationsklassen. "Durch Integration von DV-Technik, Bürotechnik und Nachrichtentechnik zu einer umfassenden Informations- und Kommunikationstechnik können auf der Basis von ISDN (»Integrated Services Digital Network«), WAN (»Wide Area Network«) oder LAN (»Local Area Network«) »multifunktionale Workstations« angeboten werden, die durch ihre Ausstattung mit Mail-, File- und Printservern prinzipielle Zugangsmöglichkeiten zu allen Informations- und Kommunikationsdiensten und die leichte und schnelle Integrationsmöglichkeit aller im

---

[156] Einen kurzen Überblick über diese Ansätze geben Paetau [Paetau 1990, S. 180ff.] sowie Hamborg und Schweppenhäußer in ihrer Expertise »Arbeits- und Softwaregestaltung« [Hamborg/Schweppenhäußer 1991].

Büro zu verarbeitenden Informationsklassen (Text, Grafik, Tabellen etc.) bieten. Auf diese Weise kann die bisherige Zersplitterung der technischen Figuration aufgehoben, sogenannte Medienbrüche in der Büroarbeit überflüssig werden." [ebd.]

Es ist sicherlich kein Zufall, daß bereits seit einiger Zeit über Organisations- und Kooperationsformen diskutiert wird, die dieses Integrationspotential optimal nutzen. Die **Ablösung einer funktionsorientierten Organisationsform durch eine geschäftsprozeßorientierte Organisation**, die heute verstärkt unter den Stichworten »Lean Production« oder »Geschäftsprozeß-Reengineering« diskutiert wird, ist die konsequente Form, moderne Informationstechnik mit einer neuen Organisation zu verbinden. Aber auch unterhalb dieser Schwelle einer umfassenden Reorganisation eines Unternehmens oder eines Arbeitsprozesses müssen grundlegende arbeitsorganisatorische Ansätze realisiert und die damit verbundenen Probleme und Konflikte gelöst werden, um einen optimalen Einsatz von Software im Kooperationszusammenhang zu ermöglichen.

Das Integrationspotential moderner Informationstechnik impliziert drei Aspekte, die in der Gestaltung von Kooperationsbeziehungen zu beobachten sind. Ursprünglich als **Organisationsmodelle ganzheitlicher Bürotätigkeiten** formuliert, wollen wir sie hier als der Technologie innewohnende Entwicklungsmomente darstellen, die erheblichen Gestaltungsbedarf im Software-Projekt erzeugen:

- die *Rückdelegation* bestimmter Unterstützungstätigkeiten an die inhaltliche Sachbearbeitung,
- die *Reintegration* durch Anreicherung von Routinetätigkeiten mit qualifizierten Aufgabenelementen und
- die *horizontale Integration* durch Zusammenführung sachverwandter Aufgaben auf gleicher Hierarchieebene. [vgl. Paetau 1990, S. 203]

Alle drei Formen bringen spezifische Probleme mit sich, zu deren Lösung das Software-Projekt beitragen muß:

Durch *Rückdelegation* werden Personen (z.B. qualifizierte Sachbearbeiter, Manager) mit Tätigkeiten konfrontiert, die sie bisher in der Hierarchie nach unten delegierten. Zum Problem wird hier weniger die mit der Rückdelegation verbundene Intensivierung der Arbeit, sondern der damit verbundene Prestige- und Motivationsverlust.

Die *Reintegration* von Tätigkeiten setzt die Qualifikation zur Übernahme dieser Tätigkeiten voraus. Dies erzeugt einen Weiterbildungsbedarf, der in vielen Unternehmen nur unzureichend gedeckt wird. Darüber hinaus verwischen die

Grenzen zwischen qualifizierter Sachbearbeitung und Hilfs-Sachbearbeitung, was in den betroffenen Abteilungen zu Interessenkonflikten führen kann.

Die *horizontale Integration* kann in die Abgrenzung der betroffenen Sachgebiete eingreifen. Einerseits sind wiederum Auswirkungen auf die notwendigen Qualifikationen der Sachbearbeiter zu berücksichtigen, andererseits können sie auch Eingriffe in die Zuständigkeiten verschiedener Abteilungen darstellen, die ebenfalls zu Konflikten führen können.

Rückdelegation, Reintegration und horizontale Integration sind eng mit dem **Entstehen neuer kooperativer Tätigkeitsstrukturen** im Arbeitsprozeß verbunden, aus denen sich neuartige Koordinationsprobleme ergeben. Der Einsatz des Computers als Kommunikationsinfrastruktur eröffnet hier neue Möglichkeiten der Kommunikation. "Wenn man unter Kommunikation nicht nur einfaches 'Mailing' oder den gemeinsamen Zugriff auf Datenbestände von verschiedenen Arbeitsplätzen aus versteht, sondern tatsächliche Abstimmungsprozesse im Sinne einer interpersonellen Situationsdeutung, dann drängt sich die Frage auf, welchen Beitrag informationstechnisch gestützte Kommunikationssysteme zur Lösung des Koordinationsproblems leisten können." [ebd., S. 207] Durch den Computer kann der Raum der Kommunikationsmöglichkeiten um neue Formen der Kommunikation erweitert werden. Genauer wäre also zu fragen, wie das Software-Projekt zur Gestaltung und Nutzung dieses Möglichkeitsraumes beitragen kann.

In der **Gestaltung des Arbeitsprozesses** zielt das Software-Projekt auf eine effektive Gestaltung und Unterstützung der Arbeit des einzelnen Menschen und der Kooperation.

- Der Mensch als Individuum ist mit bestimmten Fähigkeiten, Möglichkeiten und Grenzen ausgestattet, die für die Erreichung der Arbeitsergebnisse optimal genutzt werden sollen.
- Kooperierende Menschen, die über den Arbeitsprozeß miteinander verbunden sind, sind in ihrem sozialen Zusammenhang ebenfalls mit Leistungsmöglichkeiten und Grenzen ausgestattet, die sich erheblich auf die Qualität des Arbeitsprozesses auswirken.

In der Formulierung seiner Gestaltungsziele ist das Software-Projekt übergeordneten **Unternehmenszielen** verpflichtet, die es für seinen Gestaltungsbereich operationalisieren muß. Je nach Unternehmensphilosophie und Fähigkeit des Managements reicht die Spanne von einer diffus ausgearbeiteten Leitorientierung bis hin zu ausgefeilten, für das gesamte Unternehmen gültigen Leitlinien. In allen Fällen besteht für das Software-Projekt die Anforderung, diese Ziele entsprechend der Projektaufgabe zu interpretieren.

Die Orientierung der Arbeitsprozesse an der Erzielung hoher **Qualität** gilt als eines der wichtigsten Unternehmensziele. Unter dem Stichwort TQM (Total Quality Management) wird heute verstärkt nach Möglichkeiten gesucht, mit der Qualität der Produkte und der Kundennähe des Unternehmens Wettbewerbsvorteile zu erlangen. [vgl. z.B. Bullinger u.a. 1994, S. 26ff.] Entgegen einer früheren Qualitätssicht setzt sich dabei heute die Ansicht durch, daß Qualität nicht erprüft, sondern kontinuierlich im Produktionsprozeß erzeugt wird. Produktqualität und Prozeßqualität stellen daher bereits für viele Unternehmen die Leitlinie dar, die für alle Arbeitsprozesse im Unternehmen Gültigkeit haben soll.

Insbesondere in der Büroarbeit sind **Produkt- und Prozeßqualität unmittelbar mit dem Einsatz von Software verknüpft**. Gerade der Einsatz moderner Client-Server Architekturen scheint das Erreichen dieser Ziele zu begünstigen. So korrespondiert etwa das Integrationspotential moderner Informationstechnik mit der Orientierung an Geschäftsprozessen, die zunehmend auch in der Durchsetzung neuer Organisationsformen ihren Niederschlag finden. Moderne Formen zentraler Datenhaltung und dezentraler Bearbeitung, Objektorientierung in der Anwendungsentwicklung und die darin angelegte Transparenz und Kontrollmöglichkeit hinsichtlich der bearbeiteten Informationen stellen den Prozeß der Bearbeitung und des Austauschs von Informationen in den Vordergrund und ermöglichen hohe Qualitätsstandards in Form und Präsentation der Arbeitsergebnisse.

Das optimale Ausschöpfen dieser Gestaltungsmöglichkeiten ist heute entscheidend für Erfolg oder Mißerfolg eines Software-Projekts. Es setzt handlungsleitende Projektziele voraus, die diese allgemeinen Qualitätsziele und prozeßorientierten Strategien für den konkreten Arbeitsprozeß interpretieren.

Ansatzpunkte der Arbeitsgestaltung findet das Software-Projekt in

- den subjektbezogenen Tätigkeitsstrukturen,
- den kooperativen Tätigkeitsstrukturen,
- den sozialen Kooperationsverhältnissen

und ihrer Software-Unterstützung.

Auf der Ebene der **subjektbezogenen Tätigkeitsstrukturen** gestaltet das Software-Projekt die Arbeitsinhalte und die Regulationsbedingungen eines Individuums. Hauptaugenmerk liegt hier auf der Fähigkeit des Menschen, planende und kontrollierende Tätigkeiten effektiv in die Bewältigung seiner Aufgaben zu integrieren und damit flexibel und qualitätsorientiert zu arbeiten.

Auf der Ebene der **kooperativen Tätigkeitsstrukturen** gestaltet das Software-Projekt den Geltungsbereich der Kooperation und die Form der Koordination von

Tätigkeiten. Im Zentrum steht hier die Gestaltung der gegenständlichen Bedingungen einer flexiblen Koordination von Tätigkeiten.

Auf der Ebene der **sozialen Kooperationsverhältnisse** unterstützt das Software-Projekt die Entwicklung der sozialen Basis von Kooperationsverhältnissen. Es gestaltet die sachlichen Bedingungen der Kommunikationsstruktur durch die Erweiterung der Kommunikationsmöglichkeiten.

2 Die Analyse des Arbeitsprozesses

Grundlage der Gestaltung von Arbeit ist die Analyse des Arbeitsprozesses. Die Mitglieder des Software-Projekts müssen sich ein gemeinsames Verständnis von ihrem Gestaltungsgegenstand erarbeiten, um seine zukünftige Form bestimmen zu können. Dazu gehört zum einen, daß der betrachtete Arbeitsprozeß abgegrenzt wird. Das Software-Projekt muß bestimmen, welche Tätigkeiten und Informationsflüsse zum Arbeitsprozeß gehören und welche nicht sowie welche Rahmenbedingungen dabei zu beachten sind. Zum anderen muß der Arbeitsprozeß verstanden werden. Dies gilt nicht nur für die Entwickler im Projekt, denen der Arbeitsprozeß u.U. vollständig unbekannt ist. Auch die Anwender im Projekt müssen sich dieses Verständnis zunächst erarbeiten, denn ihr Wissen über den Arbeitsprozeß liegt häufig nur in Form unbewußten Erfahrungswissens vor, das zudem auf den eigenen Arbeitsbereich beschränkt ist. Sie müssen ihr Wissen formulieren und es erweitern, um ihren Aufgabenbereich in den Arbeitsprozeß einbetten zu können.

2.1 Unternehmensstrukturen und Arbeitsprozesse

Die Arbeit, die in einem Unternehmen geleistet wird, formiert den Gesamtarbeitsprozeß des Unternehmens. Auf dieser Ebene läßt sich der Arbeitsprozeß durch die Organisationsstruktur »Unternehmen« gegenüber dem übergeordneten gesellschaftlichen Arbeitsprozeß, in den das Unternehmen eingebettet ist, abgrenzen. Nach innen aber ist diese an organisatorischen Strukturen orientierte Abgrenzung von Teilarbeitsprozessen, wie sie das Software-Projekt zum Gegenstand hat, nicht mehr sinnvoll.

Die Organisationseinheiten, wie sie heute in Unternehmen vorgefunden werden, sind **an den Funktionen des Produktionsprozesses orientiert**. Abteilungen wie Forschung und Entwicklung, Beschaffung, Produktion, Vertrieb, Buchführung und Controlling sind in den meisten Unternehmen zu finden. Diese Funktionen wurden aus dem produktbezogenen Produktionsprozeß herausgelöst und haben sich soweit verselbständigt, daß das Wissen um ihren eigentlichen Zusammenhang weitgehend verloren ging. Demgegenüber stehen die Möglich-

keiten der modernen Informationstechnik mit ihrem Integrationspotential, welches sich in einer Neugestaltung der unternehmensinternen und -externen Informationsflüsse entfaltet. [Paetau 1990, S. 196] Sie integriert die verschiedenen informationstechnischen und kommunikationstechnischen Systeme über das gesamte Unternehmen hinweg (und darüber hinaus), bietet Zugang zu allen Informations- und Kommunikationsdiensten und hebt die bisher strikte Trennung verschiedener Informationsklassen (Text, Grafik, Tabellen etc.) auf. Unternehmensweite Datenmodelle, zentrale Datenhaltung und dezentrale Bearbeitung sind die wichtigen Innovationen im Bereich der Informationsverarbeitung, die im Software-Projekt umgesetzt werden sollen. Um diese Möglichkeiten zu nutzen und die Konsistenz der Datenbasis zu wahren, müssen ganze Informationsflüsse von ihrer Entstehung bis hin zu ihrem Endzustand gestaltet werden. Informationsflüsse und die Arbeitsprozesse, in die sie eingebettet sind, verlaufen häufig quer zu den Unternehmensstrukturen, denn sie folgen den produktorientierten Prozessen im Unternehmen und sind nicht an Funktionseinheiten orientiert.

Die Orientierung der Entwicklung von Software an diesen Strukturen, etwa einer Abteilung, konterkariert die Möglichkeiten und Effizienzvorteile, die die moderne Informationstechnologie bietet. Das Software-Projekt muß diese Einengung meiden und statt dessen genau denjenigen Arbeitsprozeß erkennen und definieren, dessen Informationsverarbeitung durch die Software neu gestaltet werden soll.

In dem dichten Netzwerk von Tätigkeiten und Ergebnissen innerhalb eines Unternehmens stellt die **Isolation eines Arbeitsprozesses** eine Abstraktion dar, die von der Berücksichtigung vieler Wechselwirkungen absieht. Die Bestimmung des Arbeitsprozesses als Gestaltungsgegenstand ist nicht durch den Einsatz einer Methode zu bewältigen. Die Herauslösung eines Teilprozesses als Gestaltungsgegenstand kann hier nur auf einer analytisch fundierten und pragmatischen Abgrenzung basieren. Nach analytischen und pragmatischen Gesichtspunkten ist diese Entscheidung in Abwägung mit der Unternehmensstrategie und des realisierbaren und wünschenswerten Nutzens der Software zu treffen. Markante Punkte dieser Bestimmung sind die eingehenden Produkte und die Ergebnisse, die der Arbeitsprozeß liefern soll. Mit ihnen sind Merkmale und Bedingungen verbunden, die die Schnittstellen des Arbeitsprozesses zur Außenwelt darstellen. Als Kriterium sollte dabei gelten, daß die Anzahl dieser Schnittstellen

relativ gering und der Zusammenhang innerhalb des Prozesses relativ hoch sein soll.[157]

2.2 Die Rahmenbedingungen des Arbeitsprozesses

Rahmenbedingungen sind gleichermaßen sowohl Einschränkungen als auch Voraussetzungen des Arbeitsprozesses. [vgl. Hacker 1986, S. 87ff.]

Der Arbeitsprozeß ist durch die **Gesellschaftlichkeit der Arbeit** bestimmt. Die Merkmale des Arbeitsprozesses ergeben sich aus den gesellschaftlichen Verhältnissen und Beziehungen, in denen gearbeitet wird.

Für das Software-Projekt haben insbesondere gesetzliche Bestimmungen unmittelbare Gestaltungsrelevanz. Sie müssen erkannt und umgesetzt werden. Andere Aspekte können sich qualitätssteigernd auswirken, indem im Software-Projekt der für den Arbeitsbereich geltende Stand von Wissenschaft und Technik reflektiert wird sowie Innovationen angeregt werden. Insbesondere die mit der Gestaltung von Arbeit verbundene Auseinandersetzung mit Werten und Normen sowie der Interessenbezug hinsichtlich der Gewichtung von Gestaltungskriterien stellen sich in jedem Software-Projekt oft als Quelle von Konflikten heraus, die die Entwicklungsarbeit maßgeblich beeinflussen.

Der Arbeitsprozeß ist abhängig von der **Technologie**, also vom Mechanisierungs- und Automatisierungsgrad der Arbeitsmittel. Um ein Arbeitsergebnis zu erreichen, können verschiedene Technologien eingesetzt werden, mit denen Basisentscheidungen über

- einzusetzende Arbeitsmittel,
- Organisationsprinzipien und
- erforderliche Arbeitstätigkeiten

getroffen werden. Mit der Technologie werden Handlungsspielräume und Handlungsnotwendigkeiten abgesteckt.

Insbesondere die nicht zur Disposition stehenden Arbeitsmittel (z.B. der alte Computer, der weiter eingesetzt werden muß, der Aktenschrank, der auf keinen Fall durch elektronische Aktenverwaltung ersetzt werden soll u.a.) setzen dem Projekt technologische Rahmenbedingungen, die der Gestaltung der Arbeit und

---

[157] Für die Abgrenzung von Organisationseinheiten werden die Kriterien der Autonomie in den Außenbeziehungen und der Kooperation in den Binnenverhältnissen verwendet. [vgl. hierzu z.B. Kubicek/ Höller 1991] Obwohl es hier i.d.R. nicht um die Bestimmung von Organisationseinheiten geht, erscheinen diese Kriterien trotzdem zur Abgrenzung eines über einen Arbeitsprozeß begründeten Systems sinnvoll.

der Software ihre eigene Gesetzmäßigkeit aufzwingen. Aber auch die Basisentscheidung für eine bestimmte IT-Architektur setzt Möglichkeiten und Grenzen der Gestaltung von Arbeit, die vorab erkannt und optimal genutzt werden müssen.

Der Arbeitsprozeß ist durch seine Organisations-Grundprinzipien bestimmt, durch die Form der **Arbeitsteilung und -kombination**. Arbeitsprozesse können nach unterschiedlichen Prinzipien organisiert sein. Diese Organisationsprinzipien betreffen

- die Beteiligung an Leitung und Planung,
- das Ausmaß der Festlegung von Vorgehensweisen,
- das Verhältnis von Einzel- und Gruppenarbeit,
- die Entlohnung sowie
- Grad und Art der Arbeitsteilung und -kombination.

Im Software-Projekt ist mit der Formulierung strategischer Ziele und der Einschätzung der Möglichkeiten und Grenzen der geplanten Software die Arbeitsteilung und -kombination in Frage zu stellen und möglichst zu optimieren.

Der Arbeitsprozeß ist von der Form der **Kooperation** abhängig. Arbeitsteilung und -kombination bestimmen die Erfordernisse und Möglichkeiten der Kooperation der am Arbeitsprozeß Beteiligten. Zu berücksichtigen sind hier die Ebenen

- Organisationseinheiten
- Arbeitsgruppen
- Individuen.

Die übergeordneten Ebenen setzen den untergeordneten Rahmenbedingungen, die bei der Gestaltung von Arbeit zu berücksichtigen sind.

Der Arbeitsprozeß ist von den **Gesetzmäßigkeiten der eingesetzten Arbeitsmittel und -gegenstände** abhängig. Insbesondere Arbeitstätigkeiten sind der Sachlogik der eingesetzten Arbeitsmittel und -gegenstände untergeordnet. Diese kann auf physikalisch/technischen, chemischen oder biologischen Vorgängen basieren. Sie äußert sich in

- Beeinflussungsmöglichkeiten,
- geeigneten Eingriffspunkten und
- wahrscheinlichen Folgen von Eingriffen

im Arbeitsprozeß. Die Gestaltung von Arbeit setzt die Kenntnis dieser Gesetzmäßigkeiten voraus.

Der Arbeitsprozeß ist von der Möglichkeit abhängig, zu **eigenständigen Zielsetzungen** zu gelangen. Diese Möglichkeit wird durch

- die Sachlogik der Arbeitsmittel und -gegenstände,
- Organisationskonzepte,
- die Funktionsteilung zwischen Mensch und Maschine,
- die Arbeitsteilung und -kombination,
- die Gestaltung der Arbeitsplätze,
- Normen und
- Arbeitsanweisungen

bestimmt. Im Software-Projekt sind diese Bestimmungsmomente zu erkennen und im Sinne einer Veränderung und Vergrößerung des Handlungsspielraums zu hinterfragen.

2.3 Die Ist- Analyse

Der unmittelbare Zugang zum Verständnis des Arbeitsprozesses ist die Tätigkeitsebene. In ihr spiegeln sich Kooperationsformen und der Einsatz von Arbeitsmitteln und -gegenständen in einer Form wider, die für die Anwender im Projekt formulierbar ist.

Die Analyse stellt für das Projekt in zweierlei Hinsicht ein Problem dar:

a) Die Auseinandersetzung mit der Ist-Situation birgt die Gefahr, sich in den Details des Arbeitsalltags zu verlieren und Modelle und Beschreibungen von einem Zustand anzufertigen, den man überwinden will. Die Kunst der Analyse besteht auch darin, rechtzeitig aufzuhören, ohne daß man wirklich alles verstanden hat.

b) Die Formulierung einer Ist-Situation geschieht immer schon unter dem Eindruck des Gestaltungsauftrages. Die Perspektive der Projektteilnehmer ist dadurch auf die Aspekte beschränkt, die sie subjektiv für gestaltungsrelevant halten. Mit der Analyse beginnt also schon die Gestaltung, die durch die Wahl der richtigen Perspektive erheblich beeinflußt wird.

Die Analyse ist eine grundlegende Aktivität des Projekts, deren Qualität nicht mit der erschöpfenden Erhebung aller Details des Arbeitsprozesses verbunden ist. Diese lassen sich im Projektverlauf schneller und zweckgerichteter erheben. Wesentlich für die Analyse ist ein grundlegendes Verständnis der verfolgten Zwecke des Arbeitsprozesses, seiner Bedingungen und der grundlegenden Schritte zur Zweckerreichung.

Hierfür ist es sinnvoll, die Tätigkeiten im Arbeitsprozeß in folgenden Dimensionen zu betrachten:

- Die **fachlich-inhaltliche Dimension**: sie ist durch den fachlich-inhaltlichen Zweck der Arbeit und die notwendig einzubringenden fachlichen Qualifikationen bestimmt. Diese Dimension ist weitgehend über die mehr oder weniger verallgemeinerten fachlichen Grundlagen (etwa Finanzbuchhaltung als eigenständiges, dokumentiertes Fachgebiet) zu erschließen.
- Die **funktional-rollenspezifische Dimension**: sie ist durch die Notwendigkeit der Koordination der Arbeitstätigkeiten im Arbeitsprozeß bestimmt. Teile der funktional-rollenspezifischen Dimension der Arbeit lassen sich über Organigramme und Arbeitsplatzbeschreibungen erschließen. Diese Quellen allein werden allerdings ein unvollständiges Bild ergeben, da die Arbeitsrealität in den Unternehmen über diese Beschreibungen hinausgeht, sie sogar ganz oder teilweise außer Kraft gesetzt hat. Notwendig ist eine Analyse vor Ort.
- Die **formal-prozessuale Dimension**: sie ist durch die Aktivitätsmuster, die zur Bewältigung der Arbeitsaufgabe führen, bestimmt. Gemeint sind Aktivitätsmuster der Informationsverarbeitung und des kommunikativen Handelns, die sowohl der fachlich-inhaltlichen Bewältigung der Aufgabe als auch der Wahrnahme der Koordinationsfunktion dienen. [vgl. Paetau 1990, S. 199ff.]

3 Die Gestaltung des Arbeitsprozesses

Das strategische Ziel von Software-Projekten liegt im weitesten Sinne in der **Effektivierung von Arbeitsprozessen**. Der Arbeitsprozeß jedoch, der sich im alltäglichen Handeln von Menschen realisiert, entzieht sich weitgehend dem Zugriff des Software-Projekts. Es ist daher zu fragen, welche Aspekte des Arbeitsprozesses zum Gestaltungsgegenstand werden können.

Die Tätigkeiten einzelner Menschen oder die Kooperation mehrerer Menschen können wir als das **Ausführen von Aufgaben unter bestimmten Handlungsbedingungen mit spezifischen Regulations- und Koordinationserfordernissen** interpretieren. Ausgangspunkt für das Arbeitshandeln von Menschen, seien es Individuen oder Gruppen, ist die Übernahme von Aufgaben. Dabei unterscheidet sich die Aufgabe als psychologische Kategorie von einem gestellten Arbeitsauftrag dadurch, daß die übernommene Aufgabe eine Interpretation des Auftrags darstellt und damit bereits die individuellen Voraussetzungen seiner Bewältigung umfaßt. Dies sind etwa das individuelle Begreifen des Auftrags, seine Bewertung, die Bereitschaft und das Können zu seiner Ausführung. [vgl. Hacker 1986, S. 69]

Aufgaben werden unter bestimmten Bedingungen der Handlungsregulation und der sozialen Kontrolle ausgeführt. [vgl. ebd., S. 34ff.] Wir wollen hier die gegenständlichen Bedingungen in Form der Beschaffenheit der Arbeitsmittel und -gegenstände sowie die sozialen Bedingungen hervorheben.

Gegenstand des Software-Projekts sind gleichermaßen die Arbeitsaufgaben und die Arbeitsbedingungen, die sowohl hinsichtlich der subjektiven Tätigkeiten als auch der Kooperation gestaltet werden müssen, um die Ziele der Arbeitsgestaltung zu realisieren.

3.1 Das Leitbild der »vollständigen Tätigkeitsstruktur«

In der Praxis des Arbeitsprozesses sind Tätigkeiten durch die Arbeitsteilung und die Arbeitsorganisation am einzelnen Arbeitsplatz bestimmt. Es lassen sich folgende Tätigkeitsklassen identifizieren:

- Zielsetzung,
- Vorbereitung,
- Ausführung und
- Kontrolle.

Je nach Charakter des Arbeitsprozesses umfaßt er Tätigkeiten der **Zielsetzung**. Diese Zielsetzung kann sich auf die Ergebnisse und Ziele des Arbeitsprozesses und auf Zwischenergebnisse und -ziele beziehen. Abgesehen von den seltenen Fällen, in denen diese in allen Aspekten vorgegeben sind, kann sich die Zielsetzung auf

- Zeitziele,
- Mengenziele und auf
- Eigenschaften des Arbeitsergebnisses

beziehen.

Bezogen auf die Zielsetzungen sind **Vorbereitungstätigkeiten** notwendig. Sie umfassen

- Ziel- und Bedingungsanalysen,
- Mittel-Weg-Ermittlungen und
- Entscheidungen über Vorgehensweisen.

Entsprechend der in der Vorbereitung entstandenen Handlungspläne wird die eigentliche Tätigkeit ausgeführt. In der **Ausführung** werden die Arbeitsgegen-

stände mit den zur Verfügung stehenden Arbeitsmitteln in Hinblick auf das angestrebte Arbeitsergebnis bearbeitet.

Im Anschluß an die Ausführung folgen **Kontrolltätigkeiten**, in denen das erreichte Arbeitsergebnis mit dem vorweggenommenen, angestrebten Arbeitsergebnis verglichen und Maßnahmen zur Qualitätssicherung festgelegt werden.

Die in der Vergangenheit etablierte Form der Arbeitsteilung, auf die das Software-Projekt in der Analyse stoßen wird, basiert auf dem Entkoppeln dieser aus der Sicht des Produkts zusammengehörigen Tätigkeiten und der funktionsorientierten Zusammenfassung. Damit entstanden funktionsorientierte Tätigkeitsstrukturen, die weitgehend auf einer dieser Klassen basierten. Produktorientierte Tätigkeitsstrukturen, also eine Kette von Tätigkeiten, die alle Schritte zur Erstellung eines Produkts umfassen, sind durch diese funktionsorientierten Strukturen überdeckt und nicht unmittelbar zu erkennen.

Auf der Ebene der Arbeitstätigkeiten führt die Forderung nach der **Integration der Qualitätskontrolle in den Arbeitsprozeß** in flexiblen Kooperationsverhältnissen und unter Einsatz moderner Software zu den Gestaltungskriterien der Vollständigkeit von Tätigkeitsstrukturen mit Freiheitsgraden.

Die Basisentscheidung für eine Integration der Qualitätskontrolle in den Arbeitsprozeß ist zunächst in der engeren Verknüpfung kontrollierender und ausführender Tätigkeiten zu sehen. Nur Tätigkeitsstrukturen, in denen Ausführung und Kontrolle in engen Rückkopplungsschleifen aufeinander bezogen sind, können die Grundlage für eine über den gesamten Produktionsprozeß sichergestellten hohen Produktqualität darstellen. Im Vordergrund steht hier weniger der Einsatz von Testmethoden zur Feststellung des erreichten Standes. Kurze Rückkopplungsschleifen sollen weit mehr die Fähigkeit des arbeitenden Menschen nutzen, den erreichten Arbeitsstand permanent mit einem vorgestellten idealen Endergebnis zu vergleichen und sich in der Ausführung auf diese »Meßergebnisse« einzustellen.

Wird die Qualitätssicht um Aspekte der Prozeßqualität wie Flexibilität und Kundennähe erweitert, so muß die Tätigkeitsstruktur neben ausführenden und kontrollierenden Tätigkeiten auch Tätigkeiten der Zielsetzung und der Arbeitsvorbereitung umfassen sowie mit Freiheitsgraden ausgestattet sein.

Flexibilität und Kundennähe sind Aspekte, die für die Unternehmen eine zunehmende Rolle spielen und die sich auf alle Arbeitsprozesse im Unternehmen auswirken. Neben anderen Effekten bringen sie v.a. mit sich, daß Zeit- und Mengenziele sowie Form und Inhalt der Produkte von Auftrag zu Auftrag sehr stark differieren können. Eine erschöpfende Vorgabe von Zielen und Ergebnissen ist also immer seltener möglich. Mit den Zielen müssen auch die Arbeits-

wege an den jeweiligen Auftrag angepaßt werden. Die Definition von Zielen und die Entscheidung für bestimmte Arbeitswege und -mittel muß also eng mit der Ausführung von Aufträgen verknüpft sein, um flexibel auf Veränderungen reagieren zu können.

Die Möglichkeiten einer selbständigen Zielsetzung und Planung werden als **Freiheitsgrade** einer Tätigkeitsstruktur bezeichnet. Je nach Ansatzpunkten eigenständiger Zielstellung können sie in unterschiedlicher Form vorkommen. Hinsichtlich der Ausprägung von Freiheitsgraden wird unterschieden

- keine Freiheitsgrade (FG) für selbständige Ziele,
- FG für zeit- und mengenbezogene Ziele (Z),
- FG für Z + die Abfolge von Verrichtungen (A),
- FG für Z + A + Verfahren und Mittel (V),
- FG für Z + A + V + zu findende Produkteigenschaften. [ebd., S. 105]

Tätigkeitsstrukturen, die Teiltätigkeiten der Zielsetzung, Vorbereitung, Ausführung und Kontrolle umfassen, werden als vollständige Tätigkeitsstrukturen bezeichnet.[158]

Im Zuge der Realisierung moderner Unternehmensstrategien und des Einsatzes moderner Informationstechnik gewinnt das **Leitbild der »vollständigen Tätigkeitsstruktur«** zunehmend an Bedeutung. Sowohl in der arbeitsplatzbezogenen Gestaltung von Aufgabenbereichen als auch in der Bestimmung der Gesamtaufgaben kooperierender Menschen ist die Vollständigkeit dieser Aufgabenbereiche durch die enge Verknüpfung planender und kontrollierender Tätigkeitsanteile eine wesentliche Grundlage der Optimierung der Handlungsregulation und der sozialen Koordination. Sie wird damit zum Gestaltungskriterium flexibler und qualitätsorientierter Arbeitsprozesse.

---

[158] Hacker bezeichnet diese als zyklisch vollständige Tätigkeitsstruktur und unterscheidet sie von der hierarchischen Vollständigkeit. Sie bezeichnet die Nutzung aller Ebenen der kognitiven Regulation von Tätigkeiten, die dem Individuum zur Verfügung stehen. Zu denken ist hier an nicht bewußt regulierte, automatisierte Vorgänge und bewußt regulierte Vorgänge, die eher wahrnehmend/vorstellend oder eher denkend (algorithmisch, nicht-algorithmisch, kreativ) ablaufen. Es wird davon ausgegangen, daß in dieser Hinsicht vollständige und umfassend fordernde Tätigkeiten für die Wahrung und Entwicklung des Qualifikationsstandes, für Motivation , Wohlbefinden und psychische Gesundheit von hoher Bedeutung sind. [Hacker 1986, S 163]

3.2 Die arbeitsplatzbezogene Gestaltung von Arbeit

Der Arbeitsprozeß realisiert sich im Handeln der Menschen, in ihren Arbeitstätigkeiten. Bezogen auf den einzelnen Menschen hat die **Arbeitstätigkeit** "folgende psychologisch relevanten Eigenschaften:

1. Sie ist bewußte, zielgerichtete Tätigkeit;
2. gerichtet auf die Verwirklichung eines Zieles als vorweggenommenes Resultat (Produkt), das
3. vor dem Handeln ideell vorgegeben war;
4. sie wird willensmäßig auf das bewußte Ziel hin reguliert;
5. bei der Herstellung des Produkts formt sich zugleich die Persönlichkeit; diese persönlichkeitsformende Wirkung ist nicht auf die Fähigkeiten und Fertigkeiten beschränkt, sondern betrifft auch den Charakter;
6. Jede Arbeitstätigkeit, auch die innerhalb der gesellschaftlichen Arbeitsteilung isoliert ausgeübte, ist in ihren wesentlichen Merkmalen gesellschaftlich bestimmt. Sie ist stets bezogen auch auf Bedürfnisse anderer Menschen und gewinnt daraus einen ausschlaggebenden Teil ihres Sinns." [ebd., S. 57]

In all ihren Bestandteilen ist die Arbeitstätigkeit der erforderlichen **Erzielung eines Resultats** untergeordnet. Dies findet in den Eigenarten der kognitiven Anforderungen ihren Niederschlag. [vgl. ebd., S. 58f.] Die ideelle Vorwegnahme des Arbeitsergebnisses impliziert nicht nur die Kenntnis seiner gewünschten Form, sondern darüber hinaus das Aufdecken von Ursache-Folge-Beziehungen und das Entwerfen von Ziel-Mittel-Beziehungen. Sinn und Erfolg bzw. Mißerfolg jedes Tätigkeitsabschnitts kann nur in Bezug auf das Endergebnis erfaßt werden. Relevante Rückmeldungen beziehen sich also auf das Resultat, weniger auf den Verlauf. Arbeitsaufgaben und die Bedingungen ihrer Ausführung müssen ständig an Situationsveränderungen angepaßt werden. Teilaufgaben können damit einen problemlösenden Charakter bekommen.

Kognitive Vorgänge befinden sich in der Arbeitstätigkeit in funktionaler Einheit mit motivationalen, willentlichen und motorischen Vorgängen. In ihrer Gesamtheit unterliegen sie der psychischen Regulation, die Hacker nach Antriebsregulation und Ausführungsregulation unterscheidet.

"1. Die **Antriebsregulation** bestimmt, ob gehandelt und welche Tätigkeit mit welcher Intensität ausgeführt wird. Zu dieser Gruppe psychischer Komponenten der Tätigkeit gehören in erster Linie die Absichten, Vornahmen oder Vorsätze einerseits und Bedürfnisse, Interessen, Gefühle, Bestrebungen oder Überzeugungen andererseits - mithin die Ziele und Motive des Menschen. (...)

2. Die **Ausführungsregulation** bestimmt, auf welche Weise gehandelt wird. Sie ist konzentriert auf die Analyse des Ziels, die Verwirklichungsbedingungen und des Verhältnisses von Zielen und Bedingungen der Tätigkeit zum Zwecke der Festlegung, des Einsatzes und der Kontrolle von bedingungsadäquaten Verfahren der Zielerreichung. Dieses notwendige Ermitteln, Einsetzen und fortlaufende Anpassen bedingungsadäquater Wege der Produktherstellung macht die Tätigkeit zur Erfüllung von Aufgaben. Zu dieser Komponente zählen daher die Wahrnehmungs-, Vorstellungs-, Denk- und Gedächtnisprozesse." [ebd., S. 70f.]

3.2.1 Die Gestaltung vollständiger Aufgabenbereiche

Die arbeitsplatzbezogene Gestaltung zielt auf die **effektive Nutzung und Unterstützung der menschlichen Fähigkeit, in sich verändernden Situationen kompetent und variabel zu handeln**. Als Bedingung dieses Arbeitsgestaltungsziels kann die Orientierung an »vollständigen Tätigkeitsstrukturen« angesehen werden.

Nach Maßgabe der Gestaltung optimaler Bedingungen der Handlungsregulation sind vollständige Tätigkeitsstrukturen als effektivste Form anzusehen. Mit ihrer engen Kopplung planender und kontrollierender Tätigkeiten an die ausführende Tätigkeit sind sie die Basis eines flexiblen und fehlerfreien Reagierens auf sich verändernde Bedingungen. Es wird davon ausgegangen, daß sie im Vergleich zu unvollständigen Tätigkeiten eher förderliche Auswirkungen auf den arbeitenden Menschen haben, indem Übermüdung, Langeweile und Dequalifizierung vermieden werden. Darüber hinaus können über das Leitbild vollständiger Tätigkeiten die technologischen Gestaltungsmöglichkeiten optimal genutzt werden. Das Integrationspotential moderner Informationstechnik ist nur unter einer Integration sachlich zusammengehörender Tätigkeiten auszuschöpfen. Die Zusammengehörigkeit von Tätigkeiten ist v.a. über den Lebenszyklus von Informationsobjekten vermittelt, in dem planende, ausführende und kontrollierende Tätigkeiten zusammengefaßt sind.

Demgegenüber führt die **Unvollständigkeit von Aufgabenbereichen** zu Effektivitätsverlusten. Sie entsteht durch

- eine sachlich ungerechtfertigte Anforderungsreduktion in der Ausführung, die einerseits zu Aktivitätsmangel und andererseits zu einer Formalisierung der Aufgabenbewältigung führt,
- das Ausklammern von Zielsetzungs- und Entscheidungsmöglichkeiten und damit der Verantwortungsübernahme,

- das Verhindern kognitiver Vorbereitungsschritte durch die Vorgabe normierter Arbeitswege,
- eine ungenügende Einbettung der Tätigkeit in den Kooperationszusammenhang, durch die die soziale Unterstützung des Individuums einerseits und die kooperative Aufgabenbewältigung andererseits behindert wird,
- die Reduktion von Disponibilitäts- und Lernanforderungen, durch die das Hinzulernen von Fähigkeiten und Kenntnissen und ihre Übertragbarkeit auf wechselnde Aufgaben behindert wird.

Zur Vermeidung dieser Gestaltungsfehler bietet das Leitbild der »vollständigen Tätigkeit« eine sinnvolle Unterstützung. Die arbeitsplatzbezogene Gestaltung von Tätigkeiten und ihre Software-Unterstützung sollte an diesem Leitbild orientiert sein und in Form projektspezifischer Gestaltungsanforderungen operationalisiert werden.

3.2.2 Die Handlungsregulation als Ansatzpunkt der Software-Unterstützung

Aufgaben und Tätigkeiten sind hierarchisch aufgebaut. Eine Gesamtaufgabe, die einer Gruppe oder einem einzelnen Menschen zugeteilt ist, besteht i.d.R. aus einer Reihe ineinander verschachtelter Teilaufgaben. Die Gesamttätigkeit, mit der diese Aufgabe ausgeführt wird, ist unterteilt in Tätigkeiten, Teiltätigkeiten, Handlungen und Operationen. [Hacker 1986, S. 72]

Alle Gestaltungsaktivitäten, die sich auf Teilaufgaben oder die Regulationsbedingungen von Teiltätigkeiten, Handlungen und Operationen beziehen, müssen in die Gestaltung der Gesamtaufgabe eingebettet werden, um sie durchgängig an den Zielen der Arbeitsgestaltung zu orientieren. Hacker schlägt folgende **Hierarchie von Gestaltungsebenen** vor:

Mit der Organisationskonzeption sowie der Arbeitsteilung und -kombination wird die Gesamtaufgabe eines Menschen oder einer Gruppe gestaltet. Die Gestaltung der **Gesamtaufgabe** sollte sich am Leitbild der »vollständigen Tätigkeitsstruktur« orientieren.

- Mit der Funktionsteilung zwischen Mensch und Computer werden **Teilaufgaben** gestaltet. Sie bestimmen die Regulationsanforderungen und sollen sich durch ihre Instrumentalität für die Gesamtaufgabe auszeichnen.
- Die Auswahl, Gliederung und Bedeutsamkeitskennzeichnung von Informationen bestimmt die aufgabenbezogene Gestaltung von **Informationsfeldern**.

- Die Kodierung, Lesbarkeits- und Identifikationssicherung (incl. der Formatisierung) bestimmt die aufgabenbezogene Gestaltung von **Einzelinformationen**.
- Die Umgebungs- und Zeitgestaltung des Arbeitsplatzes bestimmen die **Ausführungsbedingungen**. [Hacker 1987, S. 37]

Unmittelbar an die konzeptionelle Gestaltung von Aufgaben schließt sich die Frage nach einer Untergliederung in Teilaufgaben und dem **Unterstützungsbedarf durch Software** an. Die Anforderungen an Flexibilität und Variabilität von Arbeitstätigkeiten schließen die **äußere Struktur von Tätigkeiten**, also ihr zeitliche oder logischer Ablauf, als Ansatzpunkte der Gestaltung weitgehend aus. Die Formalisierung der Struktur von Tätigkeiten und ihrer Ablaufform sowie ihre Abbildung in Algorithmen und Programmen sind mit der Forderung nach vollständigen Tätigkeitsstrukturen nicht vereinbar. Vollständige Tätigkeitsstrukturen erfordern die Unterstützung durch Software.

In der Suche nach den formalisierbaren Anteilen von Tätigkeiten muß das Software-Projekt also Bereiche erschließen, die hinter der äußeren Struktur verborgen liegen: die psychischen Vorgänge, durch die Tätigkeiten reguliert werden und die einen spezifischen Unterstützungsbedarf durch Software erzeugen.

Diese psychischen Vorgänge die zur Formulierung von Zielen, zur Entscheidung für oder gegen einen Arbeitsweg und den Einsatz bestimmter Arbeitsmittel und zu Vergleichen zwischen erreichtem Stand und gewünschtem Ergebnis führen, bilden die **innere Struktur der Arbeitstätigkeit**, die sinnvolle Ansatzpunkte zur Gestaltung der software-technischen Unterstützung liefert.

Sie ist einerseits eine durch die Abhängigkeit von den Zielen der Arbeitstätigkeit bedingte inhaltliche Ordnung. Ausgehend von den gewünschten Eigenschaften des Arbeitsergebnisses finden hier Arbeitsmittel und -gegenstände und die bekannten Arbeitsverfahren ihre Bedeutung. Andererseits ist sie insofern eine funktionelle Ordnung, als sie die Regulation der Arbeitstätigkeit auf das Ziel hin organisiert.

Kennzeichnend für die innere Struktur einer Tätigkeit ist die zentrale Bedeutung ihrer **Zielgerichtetheit**. Sie kann daran gemessen werden, daß vergleichbare Vollzüge, etwa das Schreiben, sehr unterschiedliche innere Strukturen aufweisen, je nach dem Ziel des Schreibens bzw. der Bedeutung des Textes. Das Erfassen eines Diktats ist eine andere Tätigkeit als das schriftliche Entwerfen eines Referats.

Aus der Zielgerichtetheit der Tätigkeit können gestaltungsrelevante Aspekte der Orientierung und Regulierung abgeleitet werden. **Komponenten der inneren Struktur** sind

1. das Richten auf das Ziel,
2. das Orientieren an den technologischen Gesetzmäßigkeiten,
3. das Entwerfen von Handlungsprogrammen,
4. das Entscheiden für Mittel und Wege und
5. die Kontrolle durch Vergleich von Stand und Ziel. [Hacker 1986, S. 111]

Das Leitbild der Unterstützung vollständiger Tätigkeitsstrukturen erfordert ein **Umdenken in der Software-Entwicklung**. Statt der Formalisierung und Abbildung ganzer Arbeitsabläufe muß sich der Schritt der Formalisierung an der inneren Struktur von Tätigkeiten, also an den Komponenten der Handlungsregulation orientieren. Auf dieser Grundlage können kleine, flexibel einsetzbare Software-Werkzeuge entwickelt werden, die über die Zwecke der Arbeit und das Handeln der Menschen in einen sinnvollen Zusammenhang gesetzt werden. Das Software-Projekt schafft so Arbeitsbedingungen, die mit den neu formulierten Aufgaben korrespondieren und effektiv die Handlungsregulation unterstützen.

3.3 Gestaltung der Kooperation

In der Betrachtung des Arbeitsprozesses haben wir uns bisher auf das Individuum mit den Tätigkeiten, die es verrichtet, konzentriert. Im Verständnis des Arbeitsprozesses ist uns die Tätigkeit am nächsten, denn sie betrifft uns unmittelbar und fordert uns körperliche und geistige Leistungen ab.

Die individuelle Tätigkeit aber steht immer in Verbindung mit den Tätigkeiten anderer. Die Aufgabe, die wir erfüllen, ist immer Teil einer umfassenden Aufgabe einer Gruppe, eines Unternehmens oder eines noch größeren gesellschaftlichen Zusammenhangs. Es ist offensichtlich, daß wir Arbeit nicht als die Tätigkeit eines Robinson Crusoe betrachten können, der nur für sich selbst produziert, sich an keine Konventionen halten muß und niemandem verpflichtet ist. In der modernen arbeitsteiligen Gesellschaft basiert das wirtschaftliche Geschehen auf einer engen Verflechtung der Arbeit, die die Menschen in ihr leisten. Von den internationalen Wirtschaftsbeziehungen bis hin zu den kleinsten Produktionseinheiten in einem Unternehmen haben sich sehr komplexe **Kooperationsstrukturen** herausgebildet, die die Tätigkeit jedes einzelnen in ein dichtes Netz von Abhängigkeiten einordnet.

Die effektive Gestaltung der Kooperation sowie die Ableitung und Umsetzung von Gestaltungsanforderungen an die Software gehören zu den Aufgaben des Software-Projekts. Es kann dabei die Kooperationsbeziehungen auf allen Ebenen zum Gegenstand haben. National und international ist jedes Unternehmen in ein Geflecht von Lieferanten und Kunden eingebunden, auf das es sich zunehmend auch in seinen computerunterstützten Informationsflüssen bezieht. Unternehmensintern sind alle Funktionsbereiche und Sparten verbunden und werden zunehmend auch informationstechnisch integriert. Kooperationsbeziehungen über Abteilungs-, Unternehmens- und Ländergrenzen hinweg gehören zunehmend zu den Anforderungen der Software-Entwicklung.

Dabei ist der **Computer als Kooperationsmedium** nicht auf spezielle Anwendungsfelder, wie sie heute etwa unter der Bezeichnung "Groupware" zusammengefaßt werden, beschränkt. Jedes Software-System, das unterschiedliche Aufgabenbereiche und Tätigkeiten miteinander verbindet, muß als »CoopWare«, und jeder Arbeitsprozeß als kooperativer Prozeß gestaltet werden.

Bei der Gestaltung der Kooperation sollen die **kooperativen Aufgabenstrukturen und die soziale Koordination effektiver gestaltet** werden. Der Einsatz moderner IT bietet hier die Möglichkeit, die zeitlichen und räumlichen Restriktionen in der Kooperation zu reduzieren und die in unstrukturierten Aufgabenfeldern angelegte Verschränkung von Tätigkeiten zu entfalten und zu unterstützen. Damit entstehen neue komplexe Tätigkeitsstrukturen und Koordinationserfordernisse, an deren Unterstützung sich das Software-Projekt orientieren muß.

3.3.1 Die Gestaltung kooperativer Aufgabenbereiche

Ebenso wie für die arbeitsplatzbezogene Gestaltung von Aufgabenbereichen ist das **Leitbild »vollständiger Tätigkeitsstrukturen«** für die Gestaltung der Kooperation eine sinnvolle Orientierung. Auch hier stellt die enge Verknüpfung der verschiedenen Teiltätigkeiten im Arbeitsprozeß eine wesentliche Bedingung für eine effektive soziale Koordination der Arbeit und der Realisierung flexibler und qualitätsorientierter Arbeitsprozesse dar. Die Möglichkeit vollständiger Tätigkeitsstrukturen hängt hier wesentlich vom **Geltungsbereich** ab, in dem in der Kooperation Festlegungen getroffen werden können oder müssen. Hacker unterscheidet kooperative Festlegungen hinsichtlich

- allgemeiner Umgebungs- und Raumeinflüsse,
- Menge bzw. Tempo,
- Abfolgen und Zeitpunkten von Maßnahmen,
- Arbeitszeitorganisation,

- Maßnahmenfestlegungen,
- Arbeitsteilung und
- Aufgabenfestlegung. [ebd., S. 95]

In der Regel werden im Arbeitsprozeß Arbeitsergebnisse in Kooperation erzielt. Die Tätigkeiten der arbeitenden Menschen sind sinnvoll aufeinander und auf das angestrebte Resultat gerichtet. Sie bilden dabei Strukturen, deren Charakter durch

- das räumliche Verhältnis,
- das zeitliche Verhältnis und
- das sachliche Verhältnis

der Tätigkeiten des Arbeitsprozesses bestimmt ist. Diese Struktur bezeichnen wir als **kooperative Tätigkeitsstruktur**.

Die kooperative Tätigkeitsstruktur korrespondiert mit den Kooperationsformen, die im Arbeitsprozeß etabliert sind. Für die Klassifizierung der **Kooperationsformen** schlägt Hacker folgende Grobeinteilung vor:

- isolierte Einzelarbeit, die in Ausnahmefällen vorliegen kann,
- Arbeit im Raumverband,
- Arbeit im Sukzessivverband und
- Arbeit im Integrativverband. [ebd., S. 93ff.]

Die Arbeit im *Raumverband* bezeichnet die gleichzeitige Arbeit mehrerer Menschen in einem Raum. Diese Kooperationsform basiert auf einer weitgehenden Unabhängigkeit der Tätigkeiten. Kooperationsbeziehungen sind allein über die räumliche Nähe vermittelt. Zeitliche und sachliche Kopplungen der Tätigkeiten werden nicht berücksichtigt. Spezifische Leistungsmerkmale eines entwickelten sozialen Zusammenhangs können sich in dieser Kooperationsform nicht entwickeln.

In der Arbeit im *Sukzessivverband* werden Tätigkeiten nacheinander unter verschiedenen Aufgabenstellungen am gleichen Arbeitsgegenstand ausgeführt. Eine Tätigkeit wird nur nach auftragsgerechtem Abschluß der vorhergehenden Tätigkeit begonnen. Die zeitliche und sachliche Kopplung von Tätigkeiten wird extrem formalisiert und einer variablen Handlungsweise entzogen. Insbesondere die sachliche Kopplung der Tätigkeiten kann in dieser Kooperationsform nicht entfaltet werden. Wirkungen von Verzögerungen oder Mängeln setzen sich fort und können nur in einer nachfolgenden Qualitätskontrolle behoben werden.

Im *Integrativverband* sind mehrere Arbeitende gleichzeitig arbeitsteilig abgestimmt am gleichen Arbeitsgegenstand tätig. In Verbindung mit der Möglichkeit variabler Handlungsweisen und geringer zeitlicher Kopplung bietet die gleichzeitige Bearbeitung im Integrativverband die Möglichkeit, die Qualitätskontrolle in den Arbeitsprozeß zu integrieren.

Für die Gestaltung moderner Arbeitsprozesse sind neben der Integration von Tätigkeiten auf der Arbeitsplatzebene in vollständige Aufgabenbereiche folgende Gestaltungsanforderungen relevant:

- die Ausdehnung des Geltungsbereichs der Kooperation und
- die **Flexibilisierung der Tätigkeitsstrukturen** durch die Aufhebung zeitlicher und räumlicher Restriktionen sowie die Entfaltung der sachlichen Verhältnisse der Tätigkeiten.

Kooperation kann durch **räumliche Nähe und unmittelbare Zugänglichkeit** zu den Arbeitsmitteln und -gegenständen geprägt sein oder durch räumliche Distanz. Durch den Einsatz der Informationstechnik wird der Druck zu räumlicher Nähe, der durch die Eigengesetzlichkeit traditioneller Arbeitsmittel und -gegenstände ausgeübt wurde, reduziert. Bisher waren Arbeitsmittel und -gegenstände physisch nur an einem Ort vorhanden. Alle Tätigkeiten, die sich auf diese Gegenstände beziehen, müssen also dort verrichtet werden. Mit der Informationstechnik wird dieses Gesetz aufgehoben. Der Zugang zu Informationen und Software-Werkzeugen ist lediglich an das Vorhandensein eines Computers und ggf. einer Verbindung zum Speichermedium der Informationen gekoppelt. Tätigkeiten, die sich auf diese Informationen beziehen, können also im Prinzip überall ausgeführt werden. Vom Computer geht kein Druck aus, Kooperation mit räumlicher Nähe zu verknüpfen. Folgerichtig wird mit dem Einsatz von Computern in vielen Arbeitsbereichen die räumliche Nähe - etwa unter dem Stichwort der Heimarbeit - in Frage gestellt.

Kooperation kann in unterschiedlicher **zeitlicher Kopplung** erfolgen. Tätigkeiten können sequentiell angeordnet sein oder zeitlich parallel verlaufen. Welche Form vorliegt, ist meistens in einer Mischung aus organisatorischen Festlegungen und Eigengesetzlichkeiten der Werkzeuge und Materialien begründet. Einerseits entziehen sich viele Werkzeuge, Maschinen und Materialien einer gleichzeitigen Bearbeitung, andererseits macht die häufig vorliegende Abkopplung kontrollierender Tätigkeiten die Festlegung von Kontrollpunkten notwendig, die häufig nur in sequentiellen Tätigkeitsabfolgen realisierbar sind. Auch in der zeitlichen Dimension der Kooperation kann Software-Einsatz zu erheblichen Veränderungen führen, da die besondere, nicht-stoffliche Eigenschaft der Soft-

ware eine gleichzeitige Bearbeitung zuläßt. Arbeitsvorgänge, die bisher strikt hintereinandergeschaltet wurden, können nun gleichzeitig erfolgen.

Kooperation kann in unterschiedlicher **sachlicher Verschränkung** erfolgen. Tätigkeiten können weitgehend unabhängig voneinander und nur über Eingangs- und Endprodukte miteinander verbunden sein, sie können in einem klar definierten, über exakte Regeln beschreibbaren Verhältnis stehen oder reziprok aufeinander bezogen sein. Letzteres bedeutet eine gegenseitige Abhängigkeit der Tätigkeiten in der Form, daß sie permanent wechselseitig ihre Ausführungsbedingungen verändern. Beispielhaft ist hier etwa die gemeinsame Erstellung oder Überarbeitung eines Textes, die über moderne Textverarbeitungssysteme prinzipiell möglich wird. Wie in diesem Fall kann der Einsatz von Software zu einer Zunahme reziproker Kooperationsverhältnisse führen.

Die strategische Bedeutung, die der Einsatz moderner IT heute für die Erreichung der Unternehmensziele hat, basiert zu großen Teilen auf der Möglichkeit flexibler und intensiver kooperativer Tätigkeitsstrukturen, deren Software-Basis die **zeitlichen und räumlichen Barrieren traditioneller Arbeitsprozesse reduziert und die sachlichen Verhältnisse der Tätigkeiten umfassend unterstützt**.

3.3.2 Die Bedeutung sozialer Kooperationsstrukturen im Arbeitsprozeß

Bisher haben wir Kooperation als zeitliches, räumliches und sachliches Verhältnis von Tätigkeiten betrachtet. Den Menschen, der diese Tätigkeiten ausführt, haben wir weitgehend ausgeblendet. Bei dieser sehr eingeschränkten Sicht auf Kooperation dürfen wir nicht stehenbleiben, denn es ist der Mensch, der Kooperationsverhältnisse gestaltet und damit bestimmt, wie der - etwa durch Software - abgesteckte Handlungsraum genutzt wird.

Der arbeitende Mensch gestaltet Kooperationsverhältnisse nicht allein als ein Verhältnis seiner Tätigkeiten zu den Tätigkeiten anderer. Er gestaltet sie als **soziales Verhältnis**, in welches die Arbeit eingebettet wird. Seien es hierarchische Verhältnisse, Verhältnisse unter Kollegen, im Arbeitsalltag eingespielte Regeln oder per Dienstanweisung gesetzte Vorschriften, Kommunikationsmöglichkeiten und -grenzen im Kollegenkreis oder mit Vorgesetzten: soziale Verhältnisse regeln tagtäglich unsere Arbeit. Sie haben maßgeblichen Einfluß darauf, ob die Arbeit gut oder schlecht ausgeführt, mit viel Belastung oder wenig bewältigt wird.

Im sozialen Zusammenhang bekommt die Arbeit einen Auftragscharakter, mit dem Erwartungen an uns herangetragen werden. [vgl. Falck 1992, S. 158f.] Wir verhalten uns zu diesen äußeren Erwartungen, indem wir in Auseinandersetzung

mit unseren eigenen Einstellungen Motivation entwickeln, die gestellten Aufgaben zu erfüllen. Im sozialen Kontext entwickeln wir unsere Motive, Absichten, Gefühle und Überzeugungen, die als Antrieb zur Auftragserfüllung wirken und die Form und Intensität, in der wir Tätigkeiten ausführen, maßgeblich bestimmen; "d.h. wie wir die Folge unserer Handlungen wählen und mit welchem Einsatz wir handeln, mit welchen Werkzeugen und bis zu welchem Punkt wir arbeiten." [ebd., S. 159]

Im Unternehmen ist der Mensch in seinen Kooperationsverhältnissen in eine i.d.R. unternehmensweite **Organisation** eingebettet. Sie hängt über gemeinsame Ziele, Arbeitsgegenstände, Werkzeuge, über die gegenseitige Ergänzung von Fähigkeiten und über gemeinsame Bedingungen zusammen, bildet bestimmte Strukturen heraus und stellt damit die gegenständlichen und sozialen Bedingungen der individuellen und kooperativen Arbeit dar:

- Die Arbeitsteilung innerhalb eines Arbeitsprozesses ist in eine organisationsweite Arbeitsteilung eingebettet.
- Arbeitsmittel und -gegenstände werden nicht nur in dem einen, im Software-Projekt gestalteten Arbeitsprozeß eingesetzt. Sie haben ihre Bedeutung für die gesamte Organisation.
- Soziale Verhältnisse und Beziehungen sind nicht nur über die Erfordernisse einzelner Arbeitsprozesse geprägt, sondern über die umfassenden organisatorischen Strukturen.

Die Organisation präsentiert sich uns als die Summe von Arbeitsbedingungen, die die Art, wie wir unsere Tätigkeiten zu verrichten beabsichtigen und unsere Ziele erreichen wollen, beschränkt oder fördert.

Dabei haben wir keine starre, allein über formale Regeln begründete Organisation vor Augen. Vielmehr müssen wir **Organisationen** als etwas lebendiges ansehen, als ein System, das sich in beständiger evolutionärer Entwicklung befindet und gleichermaßen **eine formale und eine nicht-formale, lebensweltliche Natur** hat. Wir unterscheiden zwischen formaler und informeller Organisation, die erst in ihrer Einheit die Organisation jedes Unternehmens ausmacht. [ebd., S 159] Eine Trennung ist hier nur sinnvoll, um unterschiedliche Ansatzpunkte mit divergierenden Entwicklungsrichtungen in der Organisationsentwicklung aufzuzeigen.

Die **formale Organisation** umfaßt die Regeln und Bedingungen für den Normalbetrieb, d.h. für den Fall, daß die Arbeit ohne Störungen verläuft, und Entscheidungsregeln für normierte Störungssituationen. Hierarchische Strukturen, Organigramme, Arbeitsplatz- und Aufgabenbeschreibungen im Rahmen eines

Arbeitsteilungsmodells, Dienstanweisungen und die gegenständlichen Bedingungen der Arbeit wie Räume, Materialien und Werkzeuge sind der formalen Organisation zuzurechnen.

Die formale Organisation muß im konkreten Arbeitsablauf interpretiert und an die konkreten Bedingungen, Störungen und Abweichungen angepaßt werden. Dies leistet die **informelle Organisation**, die die formale Organisation überlagert. Sie bildet ebenfalls Regeln und Bedingungen heraus, die der situativen Reaktion auf sich verändernde Erfordernisse dient. Die informelle Organisation "vereint die sozialen Kompetenzen und das Ensemble individueller Verhaltens- und Bewältigungsstrategien, ohne die eine formale Organisation nicht »funktionieren« würde." [ebd., S 162f.]

Software gehört zur formalen Organisation. In ihr sind Regeln, Arbeitsweisen und -methoden vergegenständlicht, die Bedingungen der Arbeit darstellen. Sie kann festschreiben, wann und in welcher Reihenfolge Tätigkeiten auszuführen sind, sie kann das Verhältnis von Tätigkeiten festlegen, sie eröffnet oder verschließt Handlungs- und Entscheidungsspielräume und sie überträgt Verantwortung oder schließt von Entscheidungen aus. Im Software-Projekt ist auf der Ebene der formalen Organisation unvermeidlich ein Eingriff zu verzeichnen, indem Software in diesem Sinne immer auch als Organisationsmittel entwickelt und eingesetzt wird. Dabei bewegt sie sich immer im Spannungsfeld zwischen sinnvoller Unterstützung von Tätigkeitsstrukturen und einer restriktiven Beschränkung des Handlungsraumes.

Als Moment der formalen Organisation **wirkt Software auch auf die informelle Organisation**. Mit der Entwicklung und dem Einsatz einer neuen Software wird in einer Form in die informelle Organisation eingegriffen, die zu tiefgreifenden Störungen im Arbeitsprozeß führen kann. Viele Software-Projekte scheitern scheinbar nachträglich durch ein Phänomen, das recht vordergründig oft als Akzeptanzproblem einer Software aufgefaßt wird. Die Software wird dann gar nicht oder nicht im gedachten Sinne eingesetzt. Die Mitarbeiter versuchen, mit den bis dahin etablierten Vorgehensweisen und Verfahren ihre Aufgaben zu bewältigen, obwohl sich die Bedingungen ihrer Arbeit durch die Software grundlegend verändert haben. Wir können hier einerseits ein Qualifikationsproblem identifizieren, das einfach nachvollziehbar aus einer mangelnden Vermittlung des neuen Werkzeugs entsteht. Daraus allein aber ist dieses Phänomen nicht zu erklären, das auch bei weitgehend selbsterklärender Software und erheblichem Schulungsaufwand auftritt. Maßgeblich für diese Probleme erscheinen eher die Eingriffe in die informelle Organisation. Erfolgskritische Situationen im Arbeitsprozeß werden v.a. durch die informelle Organisation bewältigt; der Umgang mit Sonderfällen etwa, der Umgang mit Bearbeitungsstaus, Krankheit oder Urlaub

eines Kollegen, auf den alle anderen im Arbeitsprozeß irgendwie bezogen sind usw. Jeden Tag tauchen diese Situationen auf, für die sich 'unter der Hand' Verhaltensweisen eingespielt haben, an die sich alle unausgesprochen halten. Um den Erfolg der Arbeit nicht zu gefährden, besteht für die Mitarbeiter ein unmittelbarer Druck, diese Verhaltensweisen beizubehalten und auch gegen Störungen von außen durchzusetzen.[159]

Auch eine neue Software stellt eine solche Störung dar, die in der informellen Organisation irgendwie kompensiert werden muß. Allgemein gilt, daß durch Veränderungen in der formalen Organisation, also auch durch den Einsatz einer neuen Software, die etablierte informelle Organisation entwertet wird. Eingespielte Verhaltensweisen funktionieren nicht mehr, sie müssen durch neue ersetzt werden. [ebd., S. 163] Notwendig ist ein **Organisationsentwicklungsprozeß**, der viele Risiken in sich birgt.

Einerseits soll mit der Software, in die viel Zeit und Geld investiert wurde, der Arbeitsprozeß in Zukunft effektiver gestaltet werden. Dies setzt voraus, daß die Software zielgerichtet eingesetzt wird und mit neuen Strategien, Beziehungen und Umgangsweisen eine neue informelle Organisation etabliert wird, die den Arbeitsprozeß stabilisiert. Gelingt dies nicht, wird selbst ein abgeschlossenes Projekt noch nachträglich zur Investitionsruine.

Andererseits müssen auch in der Phase der Einführung einer Software weiterhin Arbeitsergebnisse produziert werden. Dies stellt eine Anforderung dar, deren Bewältigung durch das Auflösen einer etablierten informellen Organisation in Frage gestellt wird. Es besteht also ein objektiver Zwang, an alten Verhaltensweisen festzuhalten, um notwendige Arbeitsergebnisse nicht zu gefährden.

Das Festhalten an alten Verhaltensweisen aber behindert die Entwicklung neuer. Hier schließt sich der Kreis, der nur durch ein bewußtes Umgehen im Software-Projekt zu durchbrechen ist.

In der Organisation mit ihren formalen und informellen Bestandteilen sind die sozialen Verhältnisse begründet, in denen Menschen kooperative Tätigkeitsstrukturen bewältigen können. Aus der Form der Tätigkeitsstrukturen und den Anforderungen, die sie an das Handeln der Menschen stellen, müssen wir notwendige **Entwicklungen der sozialen Verhältnisse** ableiten und Ansatzpunkte

---

[159] Diesem Ziel steht entgegen, daß Arbeit immer durch Individuen mit individuellen Interessen und Motiven ausgeführt wird, die sich nicht mit den gemeinschaftlichen Zielen decken müssen. Kooperation basiert daher immer auf einer "Koalition von Individuen mit partiell übereinstimmenden sowie voneinander abweichenden Interessen und Motiven" [Falck 1992, S. 163], die sich in einem labilen Gleichgewicht befindet und immer wieder stabilisiert werden muß.

identifizieren, an denen das Software-Projekt auf eine möglichst synchrone Entwicklung beider Momente der Kooperation Einfluß nehmen kann.

3.3.3 Soziale Integration

Der Einsatz moderner IT im Rahmen moderner Unternehmensstrategien ist mit einer Intensivierung und Flexibilisierung der kooperativen Tätigkeitsstrukturen in Arbeitsprozessen verknüpft. Diese kooperativen Tätigkeitsstrukturen sind nur in Organisationen zu realisieren, deren soziale Struktur an die resultierenden Koordinationserfordernisse angepaßt ist.

In der Regel muß heute im Software-Projekt davon ausgegangen werden, daß die bestehende Organisation die Neugestaltung kooperativer Tätigkeitsstrukturen konterkariert. Sie übt einen erheblichen **Anpassungsdruck** auf die neu entstehenden Kooperationsverhältnisse aus, der auf einer Übernahme und software-technischen Abbildung bereits bestehender Organisationsstrukturen drängt. Das Software-Projekt befindet sich also im Spannungsfeld dieses Anpassungsdrucks und der Notwendigkeit, den Arbeitsprozeß auf der Basis der Möglichkeiten moderner Software-Architekturen zu optimieren.

Der Anpassungsdruck, der von der bestehenden Organisation ausgeht, muß dabei nicht vom Auftraggeber gewünscht sein. Wir finden ihn heute in Software-Projekten in sehr unterschiedlicher Ausprägung: von der vertraglichen Festschreibung, die ganz andere Umgangsweisen erfordert als die hier beschriebenen, bis hin zu dem Beharrungsvermögen, das etablierte Verhältnisse auch entgegen anderslautenden Zielen Veränderungen entgegenbringen können. Wir haben schon dargestellt, daß die mit Software-Projekten einhergehenden Veränderungen sehr tiefgreifender Natur sind, die immer Risiken in sich bergen und mit denen verantwortungsvoll umgegangen werden muß.

Für das Software-Projekt bedeutet dies, daß es mit den ihm zur Verfügung stehenden Mitteln einen **Organisationsentwicklungsprozeß** einleiten muß, der auf eine effektive Einbettung der neu entstehenden kooperativen Tätigkeitsstrukturen in die sozialen Verhältnisse zielt.

Software-Projekte treffen dabei heute auf Unternehmen, deren Organisation einem **bürokratischen Organisationsmodell** entsprechen.[160]

---

[160] Im Vergleich von Organisationsmodellen werden häufig dichotomische Gegenüberstellungen gewählt. In der Literatur wird z.B. zwischen mechanistischer und organischer Organisationsstruktur, zwischen bürokratischen und handwerklichen/professionellen Organisationsmodellen und zwischen Mißtrauens- und Vertrauensorganisationen unterschieden. Einen Literaturüberblick geben Kubicek und Höller. [Kubicek/ Höller 1991, S. 151]

- Sie legen das Schwergewicht auf die formale Organisation in Form von starren Hierarchien, hohen Anteilen formaler Regelungen und ausgeprägter Genauigkeit formaler Regelungen.
- Sie sind funktionsorientiert strukturiert,
- Die Arbeitsteilung ist stark ausgeprägt.

Eine **funktionsorientierte Ordnung** bemüht sich um eine weitgehende Strukturierung und Algorithmisierung der Aufgabenbereiche. Im Verbund mit einer tiefen Arbeitsteilung, die immer kleinere Tätigkeitsanteile aus dem Arbeitsprozeß herauslöst und einkapselt, ist eine Organisation entstanden, die sich bemüht, Arbeit allein über **formale Regeln** zu steuern. Dienstanweisungen und Arbeitsplatzbeschreibungen stellen Verhaltensregeln für den Normalfall und für vorhersehbare Störungen auf, die weitgehend mechanistisch ohne weitere Verständigung durch Kommunikation ausgeführt werden können. Informelle Organisation gilt hier als unsicher und nicht effektiv. Ihr wird in diesen Unternehmen nur eine geringe Bedeutung zugemessen. **Variable Verhaltensweisen** sind in dieser Organisation kaum möglich und auch nicht erwünscht. Entsprechend dominiert die sachbezogene Kommunikation innerhalb der Dienstwege. Vor allem in vertikaler Richtung der Hierarchie zeichnet die Kommunikation die formale Organisation nach. Diese Organisationen sind ausgerichtet auf die Erreichung von Zeit- und Mengenzielen in Arbeitsprozessen, die weitgehend strukturierbar und wenig störanfällig sind. Probleme tauchen hier in den Situationen auf, in denen nicht vorhergesehene Störungen auftreten, die nur durch ein flexibles und kompetentes Verhalten der Mitarbeiter bewältigt werden kann. Hier stellt sich die kaum entwickelte informelle Organisation als Nachteil heraus, denn mangelnde Reaktionsfähigkeit kann aus an sich kleinen Problemen unüberschaubare Risiken machen.

Dieser Organisation stehen die Anforderungen entgegen, die das Software-Projekt mit dem zu gestaltenden Arbeitsprozeß erzeugt. Bisher getrennte Tätigkeitsanteile werden integriert, sowohl auf der Ebene des Individuums als auch auf der Ebene von Gruppen; in Folge wird die funktionsorientierte Ordnung der Unternehmen tendenziell aufgebrochen und Arbeitsteilung wird reduziert. Kooperative Tätigkeitsstrukturen werden räumlich und zeitlich flexibilisiert und sachlich intensiviert.[161] Flexibilität, Verantwortlichkeit und entwickelte Kommuni-

[161] Folgt man Galbraith [Galbraith 1977], dann sind in diesen kooperativen Tätigkeitsstrukturen insbesondere reziproke Interdependenzen von Bedeutung. Sie erfordern neben der fortlaufenden Kommunikation eine "Interpunktion des mitgeteilten Ereignisstromes, so daß Eingriffe wechselseitig möglich und erwartbar sind und Konstellationen doppelter Kontingenz, also wechselseitige Handlungsblockaden vermieden

kationsstrukturen müssen hier also als Bedingungen des Funktionierens kooperativer Tätigkeitsstrukturen angesehen werden.

Bürokratische Organisationen sind an diese Anforderungen moderner Arbeitsprozesse nicht angepaßt. Das Software-Projekt steht deshalb heute vor der Anforderung, diese Restriktionen zu durchbrechen und zur Entwicklung einer Organisationsform beizutragen, in der es gelingen kann,

- die am Arbeitsprozeß beteiligten Personen auf gemeinsame Ziele zu verpflichten, die die Grundlage der Motivation und der Handlungsorientierung darstellen,
- die gemeinsame Nutzung knapper Ressourcen durch Austausch oder gleichzeitige Nutzung zeitnah zu steuern,
- Einzelhandlungen situationsabhängig gemäß offener gemeinsam vereinbarter Konventionen zu koordinieren und
- einen permanenten Verständigungsprozeß über Ziele und Konventionen der Zusammenarbeit, die zeitnah veränderten Bedingungen angepaßt werden müssen, zu initiieren.[162]

Grundlegende Bedingungen dieser **Leistungsfähigkeit einer Organisation** ist eine flexible informelle Organisation mit einer dichten Kommunikationsstruktur.

Das Software-Projekt führt zu einer Neugestaltung der Tätigkeiten des Arbeitsprozesses und der formalen Organisation, in die der Arbeitsprozeß eingebettet ist. Tätigkeiten sind in ihrem zeitlichen, räumlichen und sachlichen Verhältnis zueinander neu bestimmt und formale Regelungen werden neu formuliert, um die Tätigkeiten zu koordinieren und den Einsatz der neuen Arbeitsmittel und -gegenstände zu regeln.

Beides führt zu einem Veränderungsdruck auf die informelle Organisation. Mit der Entwicklung der informellen Organisation muß es gelingen, die formale Organisation zu interpretieren und neue Regelungen für nicht formalisierte Situationen zu entwickeln.

---

werden können." [Wehrsig/Tacke 1992, S. 224] Demgegenüber sind sequentielle Interdependenzen, die im schrittweisen Nacheinander abgearbeitet werden können, weniger kommunikationsproblematisch. [ebd.]

[162] Diese Kriterien werden häufig zur Definition von Gruppenarbeit herangezogen. [vgl. hierzu z.B. Piepenburg 1991, S. 82ff.] Im Zuge moderner Unternehmensstrategien und dem vermehrten Einsatz der Informationstechnik bekommen sie auch in anderen, nicht explizit gruppenorientierten Arbeitsformen eine zunehmende Bedeutung.

Die informelle Organisation entsteht über die Kommunikation der Organisationsteilnehmer, die direkter Gestaltungsgegenstand des Software-Projekts ist. Zum einen ist Software-Entwicklung heute zunehmend mit der Entwicklung geeigneter Kommunikationsmedien verbunden, zum anderen werden Kommunikationsmöglichkeiten im Arbeitsprozeß durch die Gestaltung der Software-Werkzeuge erheblich beeinflußt.

Für das Software-Projekt besteht die Anforderung, die Rahmenbedingungen der Kommunikation im Arbeitsprozeß optimal durch eine **Bereitstellung angepaßter Kommunikationsmedien** und einer **kommunikationsorientierten Entwicklung von Software** zu gestalten.

Mit der Neugestaltung von Tätigkeiten und der formalen Organisation verändern sich die Zwecke, denen Kommunikation im Arbeitsprozeß dient.

Leontjew (1980) sieht als Beweggründe der **Kommunikation**:

- die Koordinierung von Tätigkeiten während ihrer Ausführung oder in vorausgehender Absprache (sachbezogene Kommunikation),
- die Übermittlung von Nachrichten, die den gemeinsamen Kontext betreffen (reine, sozialbezogene Kommunikation),
- die Klärung und Erhaltung sozialer Beziehungen (interpersonelle, modale Kommunikation als spezielle sozialbezogene Kommunikation). [vgl. Falck 1992, S. 160]

Mit der Entwicklung vollständiger Tätigkeitsstrukturen und einem professionellen Organisationsmodell verändern sich die **Kommunikationserfordernisse** gegenüber der Ausgangssituation. Die sachbezogene Kommunikation wird ausgedehnt. Stand bisher die koordinierende Kommunikation ausführender Tätigkeiten im Vordergrund, so muß sie sich heute darüber hinaus auf die Zielsetzung, die Vorbereitung und Kontrolle im Arbeitsprozeß beziehen. Vorausgehende Absprachen etwa in Form einer Arbeitsanweisung durch den Vorgesetzten nehmen tendenziell ab und werden durch zeitnahe Kommunikation ersetzt.

Die Ausdehnung der sachbezogenen Kommunikation führt zu einer Komplexität, die nur durch eine Einbettung in eine entwickelte sozialbezogene Kommunikation zu bewältigen ist. Die Organisationsteilnehmer benötigen einen permanenten Austausch über den gemeinsamen Kontext. Neu auftretende Situationen müssen interpretiert und zu einer gemeinsamen Einschätzung verdichtet werden. Der Wahrheitsgehalt von Aussagen über einen Zustand und die Regeln, die sich in der gemeinsamen Arbeit herausgebildet haben, müssen gemeinsam überprüft und ggf. verändert werden.

Diese Kommunikation kann nur in einem stabilen Geflecht sozialer Beziehungen der Organisationsteilnehmer existieren. Sie wird also eingebettet in intensive Kommunikation zu Klärung und Erhalt der sozialen Beziehungen.

Das **Software-Projekt** bewegt sich hier in drei Einflußsphären:

1. Es bestimmt die Zwecke des Arbeitsprozesses, die wiederum die Kommunikation spezifizieren.
2. Es entwickelt zentrale Werkzeuge des Arbeitsprozesses, die mit ihren möglichen Interpunktionen aufgabenangemessene Kommunikationsmöglichkeiten eröffnen oder verschließen.
3. Es entwickelt Kommunikationsmedien, die die Kommunikationsmöglichkeiten erweitern können, aber auch eine Reihe von Problemen für die Entwicklung der Kommunikationsstruktur aufwerfen.

Bei entsprechender Gestaltung kann der **Einsatz des Computers im Arbeitsprozeß die Kommunikationsmöglichkeiten erweitern**, indem er die sachlichen, zeitlichen und räumlichen Bedingungen von Kommunikation verändert.

Je nach Situation kann unter neuen Formen der Kommunikation gewählt werden. [Maaß 1991, S. 12] Zu unterscheiden sind

- synchrone und asynchrone Kommunikation (z.B. bei Einsatz von E-Mail oder Videokonferenzen) und
- synchrone oder asynchrone Bearbeitung gemeinsamen Materials.

Dieser Erweiterung der Kommunikationsmöglichkeiten stehen eine Reihe von Problemen gegenüber, die mit der Computerunterstützung von Kommunikation verbunden und im Software-Projekt möglichst vermieden werden sollten. Sie resultieren aus der Tatsache, daß die Kommunikation mit Eigenschaften verknüpft ist, die in ihrer Ausprägung bisher auf die direkte zwischenmenschliche Kommunikation ausgerichtet waren. Weder sind bisher die computergestützten Medien an diesen Eigenschaften ausgerichtet, noch hat sich der Mensch auf die erweiterten Möglichkeiten eingestellt.

Herrmann sieht folgende **Probleme der Computerunterstützung** des Kommunikationsprozesses:

1. Kontextverlust

Indem die räumliche und zeitliche Bindung der Kommunikation aufgehoben wird, geht ein wesentlicher Bestandteil des gemeinsamen Kontexts verloren. Die Absicherung der Verständigung wird erschwert.

2. Informationsüberflutung und neue Konventionen

Insbesondere im E-Mail-Bereich ist das Phänomen der Informationsüberflutung zu registrieren. Eine unüberschaubare Menge von Nachrichten geht ein, die nicht mehr adäquat bearbeitet werden kann. Es besteht die Gefahr, daß wichtige Mitteilungen übersehen werden.

3. Einschränkung und Formalisierung von Ausdrucksmitteln

Die nonverbale Kommunikation wird extrem eingeengt. Formalisierung von Kommunikation etwa über formularorientierte Nachrichtenübermittlung schränkt die Flexibilität und damit die Leistungsfähigkeit der Kommunikationsstruktur ein.

4. Die Ambivalenz von Transparenz und Kontrolle

Durch den Einsatz moderner Informations- und Kommunikationstechnik ist es möglich, im Arbeitsprozeß ein Maß an Transparenz zu erreichen, das sich in einem Zielkonflikt zu anderen Interessen befindet. "Die Vermittlung von Kommunikation und Kooperation über computerbasierte Systeme ermöglicht es, den Fortschritt arbeitsteilig verschränkter Aufgabenbewältigung detailliert zur Kenntnis zu nehmen, was eine flexible Abstimmung ermöglichen kann. (...) Diese erhöhte Transparenz impliziert, daß etwa zur Kenntnis genommen werden kann, wie schnell bzw. unterschiedlich schnell die Mitglieder einer Gruppe auf eine Nachricht reagieren. Diese Möglichkeit kann auch als Erhöhung des Kontrollpotentials interpretiert werden." [Herrmann 1991a, S. 77]

Mit diesen Problemen entsteht in der Unterstützung der Kommunikation ein **Regelungsbedarf**, in den die Entwicklung und der Einsatz computerunterstützter Kommunikationsmedien eingebettet werden muß. [vgl. Herrmann 1991, S. 63]

- Potentielle Kommunikationssituationen müssen hinsichtlich Möglichkeit und Notwendigkeit direkter und elektronisch vermittelter Kommunikation untersucht werden.
- Die angemessene Form der Computerunterstützung muß bestimmt und die zugrundeliegenden Dienste ausgewählt werden.
- Die Darstellungsformen (Text, Bild, gesprochene Sprache) müssen aufgabenabhängig bestimmt werden.
- Synchrone/asynchrone Kommunikationserfordernisse sind zu identifizieren.
- Die Erreichbarkeit von Teilnehmern muß geregelt und ggf. eingeschränkt werden.
- Filter für eingehende Nachrichten müssen zur Verfügung gestellt werden.
- Der Informationszugang muß geregelt werden.

- Die Bedingungen, unter denen das Gelingen einer Nachrichtenübermittlung transparent gemacht wird, müssen geregelt werden.
- Entscheidungen über die Form der Offenlegung des Status der Teilnehmer müssen getroffen werden.
- Entscheidungen über die Formalisierung von Nachrichtendokumenten müssen getroffen werden.

Die Gestaltung und softwaretechnische Unterstützung der Kommunikationsstruktur muß in einen **Prozeß der sozialen Integration** eingebettet sein, in dem es gelingt, die am Arbeitsprozeß beteiligten Menschen aufeinander und auf die gemeinsam verfolgten Ziele und Zwecke der Arbeit zu beziehen.

Neben der Integration flexibler, den verschiedenen Kommunikationserfordernissen angepaßter Kommunikationsmedien in das Informationssystem kann das Software-Projekt zu dieser sozialen Integration beitragen, indem

- es einen Verständigungsprozeß der betroffenen Organisationsteilnehmer initiiert;
- diesen Prozeß mit den notwendigen Informationen über den zukünftigen Arbeitsprozeß unterstützt, die eine Auseinandersetzung mit der Kooperation, den eigenen Aufgaben und der technischen Basis im Arbeitsprozeß ermöglicht;
- die formale Organisation überprüft, indem behindernde formale Regelungen ersetzt und unterstützende neue Regelungen entwickelt werden;
- in der Software Regeln implementiert werden, die zu einer Erweiterung der Handlungsmöglichkeiten und einer Unterstützung im Arbeitsprozeß führen und möglichst wenig restriktive Bedingungen schaffen.

4 Zusammenfassung

Die Orientierung der Software-Entwicklung an dem Leitbild unterstützender Werkzeuge führt in neuer Form zu einer Verknüpfung der Software-Entwicklung mit der Neugestaltung von Arbeitsprozessen. War die Software-Entwicklung bisher an der Abbildung ganzer Handlungsabläufe orientiert, so führt das Leitbild der Unterstützung zu einer Orientierung an sehr viel kleineren Einheiten menschlicher Arbeit, die der Handlungsregulation und der sozialen Koordination zuzuordnen sind. Die Verknüpfung abgebildeter Operationen und lebendiger Arbeit erfolgt flexibel im konkreten Arbeitsprozeß und ist nicht technisch reglementiert. Die Optimierung des Arbeitsprozesses ist nur mit integrierter Gestaltung lebendiger Arbeit und der Software zu leisten.

Ansatzpunkte bieten hier die Handlungsregulation und die Formen der sozialen Koordination von Arbeit, die einerseits einer Formalisierung erschlossen und andererseits als Bestandteile der lebendigen Arbeit gestaltet werden müssen.

IV Software-Entwicklung als kooperative Gestaltungsaufgabe

In diesem Kapitel wird die Software-Entwicklung als kooperative Gestaltungsaufgabe dargestellt. Um IT-Fachkräften dazu zu verhelfen, sich kompetent in den Kooperationsbeziehungen von Software-Projekten zu bewegen, ist die Vermittlung von **Kooperationskompetenz** erforderlich. Kooperationskompetenz im Software-Projekt von der Seite der Entwickler her gedacht, meint die Fähigkeit zur kooperativen Gestaltung eines Arbeitsprozesses (unter Verwendung von Software-Produkten) mit Anwendern. Diese wiederum basiert auf der Fähigkeit zur Kommunikation, beinhaltet die Kompetenz, gemeinsame Lernprozesse mit Anwendern zu initiieren und zu moderieren sowie die Lösung der Konflikte im Projekt aktiv zu unterstützen.

Das Ziel dieses Kapitels ist die Vermittlung von **Kooperationskompetenz** an IT-Fachkräfte. Dies erfordert **Wissen** über das Wesen und die Struktur von Kooperationsprozessen in Software-Projekten und Kenntnisse über die Handhabung von Methoden zur Unterstützung dieser sozialen Beziehungen. Weiterhin ist die Vermittlung **sozialer Kompetenzen** erforderlich, um in der Zusammenarbeit mit Anwendern Lernbeziehungen anregen und moderieren zu können und um Konflikte im Projekt erfolgreich zu bewältigen. Übergreifend ist ein **ganzheitliches Verständnis** der kooperativen Gestaltungsaufgabe notwendig.

1 Einleitung

Der Erfolg moderner Software-Projekte ist in hohem Maße von den Kooperationsbeziehungen der unmittelbar und mittelbar Beteiligten abhängig. Die Qualität der Beziehungen zwischen Kunden und Herstellern sowie insbesondere des Verhältnisses zwischen Entwicklern und Anwendern bestimmen in vorderster Linie die Qualität einer entwickelten Software.

Der renommierte US-amerikanische DV-Autor und Unternehmensberater Tom DeMarco kommt auf der Basis der Analyse von über 500 Software-Projekten in den USA zu folgenden Ergebnissen:

- jedes sechste Projekt wird ohne jegliches Resultat abgebrochen;
- die abgeschlossenen Projekte überziehen den Zeit- und Kostenrahmen um 100 bis 200 %;
- auf hundert Programmierzeilen ausgelieferter Software kommen im Schnitt drei Fehler. [VDI Nachrichten 46/1990, S. 34]

Die Ursachen für diese Qualitätsprobleme sind vielfältiger Natur: Die unzureichende Aufwandsabschätzung zu Beginn der Software-Projekte, die ungenügende Berücksichtigung der Wünsche und Anforderungen von Kunden und Benutzern hinsichtlich der zu erstellenden Produkte und die schlechte Teamarbeit innerhalb der Projektgruppen werden an vorderster Stelle genannt. Im Mittelpunkt dieser Probleme stehen nach Auffassung von Tom DeMarco immer soziale Aspekte: "Kein einziges Software-Projekt ist an technischen Schwierigkeiten gescheitert. (...) Die meisten Probleme sind nicht technischer, sondern soziologischer Natur." [VDI Nachrichten 46/1990, S. 34] Zu einer vergleichbaren Einschätzung der Bedeutung von Kooperationsbeziehungen für den Projekterfolg kommen auch die empirischen Untersuchungen von Weltz und Ortmann [Weltz/Ortmann 1992] und Brodbeck und Frese [Brodbeck/Frese].

Zu fragen ist: Worin ist die zentrale Bedeutung der Kooperationsbeziehungen für den Erfolg von Software-Projekten begründet?

Die Bedeutung der Kooperation für den Erfolg von Software-Projekten ergibt sich allgemein betrachtet daraus, daß Software-Entwicklung stets auf die Gestaltung eines Arbeitsprozesses zielt. Daraus resultiert eine Verquickung zwischen der zu entwickelnden Software und dem zu gestaltenden Arbeitssystem auf folgenden Ebenen:

- Die Software-Entwicklung baut auf den Besonderheiten des konkreten Arbeitsprozesses auf. Die Spezifik der jeweiligen Anwendungssituation ist der Ausgangspunkt des Software-Projektes.
- Die Besonderheiten der konkreten Arbeiten, die hier erledigt werden, begleiten die Software-Entwicklung als zentrale Gestaltungsdimension über alle Phasen hinweg.
- Der Erfolg oder Mißerfolg des Projektes mißt sich letztlich daran, ob es gelungen ist, das Arbeitssystem im angestrebten Sinne zu verbessern.

Aufgrund dieser Verquickung von zu gestaltendem Arbeitssystem und zu entwickelnder Software ist das Verstehen der Arbeit als Gestaltungsgegenstand wesentliche Voraussetzung für die Qualität der Software. Das Verstehen der konkreten Arbeit erfordert von den Entwicklern ein »Hineingehen« in die Realität, das durch eine funktionale Analyse nicht erreicht werden kann.[163] Dieses Hineingehen ist ohne die aktive Unterstützung der Anwender in einer gemeinsamen Kooperationsbeziehung nicht möglich. Die Entwickler sind darauf ange-

---

[163] "Das Verständnis der sozialen Beziehung »Arbeit« kann durch eine Daten- und Steuerflußanalyse nicht ersetzt werden; es gibt keine feuersichere »Brandmauer« zwischen Programmierung und Kontext der Anwendung (...)." [Coy 1992b, S. 6]

wiesen, daß die Anwender ihnen einen Zugang zu ihrer Realität erschließen helfen. Während vom Entwickler gefordert ist, in die Realität des konkreten Arbeitssystems hineinzugehen, steht für die Anwender im Vordergrund, Zugänge zum Verständnis der konkreten Arbeitssituation zu schaffen.

Die bei der Software-Entwicklung auftauchenden Probleme lassen sich dabei meist nicht anhand objektiver Kriterien lösen, sondern erfordern vielmehr den sozialen Austausch der Beteiligten. Sell/Fuchs-Frohnhofen [1993] unterscheiden in Anlehnung an Dörner [1976] »dialektische« von »synthetischen« und »analytischen« Problemen. Sie verweisen darauf, daß die Mehrzahl der Problemstellungen in der Software-Entwicklung dialektischer Natur sind. Bei solchen können zu Beginn des Problemlösungsprozesses weder die Ist- und Sollkriterien, noch die Operatoren, die zu einer erfolgreichen Problemlösung führen, eindeutig bestimmt werden. Das Fehlen eines »one-best-way«, der sich objektiv bestimmen läßt, macht die gemeinsame Suche nach Lösungen unter bewußter Verschränkung der jeweiligen Standpunkte und Perspektiven der einzelnen notwendig. Software-Entwicklung ist demnach mit Floyd [1989] als "multiperspektivische Realitätskonstruktion" zu verstehen. In dieser Notwendigkeit zur Verschränkung unterschiedlicher Sichtweisen liegt die Notwendigkeit für die Kooperation zwischen Entwicklern und Anwendern in modernen Software-Projekten begründet.

Naur hebt die Bestimmung der Softwareentwicklung als intellektuellen und geistigen Prozeß der Auseinandersetzung mit einem gegebenen Realitätsbereich hervor. Er bezeichnet diesen Vorgang als Theoriebildung und grenzt diese Sichtweise gegen die traditionelle Sicht ab, die Programmentwicklung als Fertigung oder Programmformalisierung nach vorgegebenen Regeln ansieht. [Naur 1985] Reisin greift diese Auffassung des Software-Entwicklungsprozesses auf. Sie verweist auf die zentrale Bedeutung der »Zweckantizipation« bei der Werkzeugherstellung und faßt dies als Prozeß der gemeinsamen Entwicklung einer Theorie über die künftigen Gebrauchszwecke der Software-Produkte. [Reisin 1992, S. 93f.] Dabei verdeutlicht die Autorin, daß die Software-Entwicklung in dieser Sichtweise kein Akt individueller Genialität ist, sondern ein **gemeinsamer Lernprozeß**.

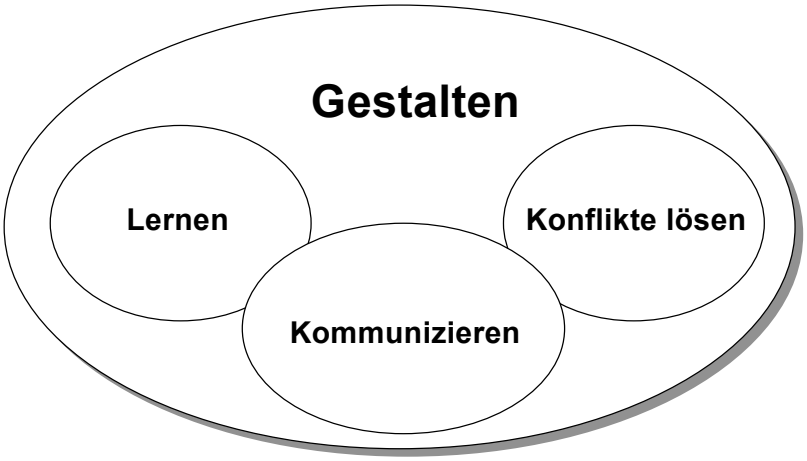
Diese gemeinsamen Lernprozesse basieren ihrerseits auf dem Austragen von Konflikten und Kontroversen der Projektbeteiligten. In diesem Sinne sind **Konflikte** die Grundlage für den Gestaltungsprozeß als kooperativer Lernprozeß.

Kommunikationsprozesse sind für das Gelingen der Software-Entwicklung von grundlegender Bedeutung. Für die Kooperationsbeziehungen im Projekt ist die **Kommunikation** von besonderer Bedeutung, weil alle sozialen Beziehungen in

erster Linie über Sprache vermittelt werden. Die gemeinsamen Lernprozesse zwischen Entwicklern und Anwendern basieren auf der Kommunikationsbeziehung zwischen den beiden Gruppen. Die Regulierung der Konflikte im Projekt ist nur auf der Basis sprachvermittelten Handelns denkbar.

Kooperative Gestaltung ist demnach durch die Dimensionen »Gestalten«, »Lernen«, »Konflikte lösen« und »Kommunizieren« gekennzeichnet.

Kooperative Gestaltung



Als **»Gestalten«** fassen wir alle Tätigkeiten im Projekt, die auf eine Veränderung des Arbeitsprozesses oder einzelne seiner Aspekte gerichtet sind. Hierin ist vorrangig die Einwirkung der Projektbeteiligten auf die »äußere Welt« angesprochen. Der Begriff umfaßt also sowohl die Herstellung technischer Artefakte als auch die Veränderung organisatorischer Strukturen oder Aufgabenzuschnitte (in Software-Projekten unter Zuhilfenahme von Software-Produkten, auch wenn das nicht zwingend ist und häufig sogar gerade nicht notwendig). Gestalten impliziert einen gemeinsamen Lernprozeß zwischen Entwicklern und Anwendern. Dieser basiert auf der Kommunikation zwischen den Projektbeteiligten und findet vor dem Hintergrund zu lösender Konflikte statt.

Als **»Lernen«** fassen wir alle Tätigkeiten im Projekt, die auf eine Erweiterung der verfügbaren Fähigkeiten hinauslaufen. Wir unterscheiden hier nach individuellen und nach interindividuellen oder kooperativen Lernprozessen. »Lernen« meint demnach die Prozesse, die »in den Individuen« ablaufen und (meist) durch das sich Aufeinanderbeziehen der Projektbeteiligten über Kommunikation zustande kommen.

»Konflikte lösen« umfaßt die Tätigkeiten, die darauf gerichtet sind, die für die Erbringung der Projektzielstellung oder die Erhaltung des Projekts als sozialem System notwendige Übereinstimmung zwischen den Beteiligten sowie im Verhältnis zu den Interessen einflußreicher Personen aus dem Projektumfeld herzustellen. »Kooperatives Lernen« sowie »Gestalten« ist zwingend mit dem »Lösen von Konflikten« verbunden. Gestalten bedeutet die Interessensphären im Unternehmen bearbeiten und erfordert daher die Lösung dieser Konflikte. Obwohl sich das »Konflikte lösen« in der Realität nicht von den anderen Dimensionen der Kooperation trennen läßt, halten wir es dennoch für sinnvoll, diese Seite der Kooperation gesondert zu behandeln, um damit die spezifische Problematik des Lernens, Gestaltens und Kommunizierens in Software-Projekten erfassen zu können. Diese Handlungen vollziehen sich nämlich immer in einer spezifischen Konfliktsituation, die von den Interessendivergenzen im Projekt und denen des Projektumfelds bestimmt ist. Die Projektbeteiligten sind daher stets gezwungen, über Konsens oder Macht eine ausreichende soziale Grundlage für die Gestaltung zu erhalten.

»Kommunikation« hat bezogen auf die drei vorher genannten Aspekte der Kooperation (Lernen, Konflikte lösen, Gestalten) eine »Querschnittsfunktion«. Sie dient im Projekt der Vermittlung der Projektaktivitäten mit der Projektumwelt, der Koordinierung der einzelnen Aktivitäten der Beteiligten und der Schaffung und Verstetigung der Identität der Gruppe.

Alle genannten Dimensionen der Kooperation sind nur analytisch, jedoch nicht empirisch zu trennen. Sie stellen jeweils Charakteristika des Software-Projektes als Kooperation dar. Durch die analytische Trennung ist es möglich, die Kooperationsprozesse im Software-Projekt theoretisch so zu erfassen, daß ein handlungskompetentes Sich-Bewegen in diesen Prozessen erlernbar wird.

Herkömmliche Projektauffassungen fokussieren die Betrachtung des Prozesses auf das Ergebnis; die Erzeugung eines Produktes. Dabei orientieren sie sich in eingeengter Perspektive lediglich an der Erstellung des Software-Produktes und dessen logischer (nicht realer!) Entwicklungsfolge. Alle Handlungen, auch die Kooperationshandlungen im Projekt, werden dann in ihrem funktionalen Verhältnis zu diesen Entwicklungsschritten als zweckrational motivierte Handlungsfolge bestimmt und diesem Ziel logisch nachgeordnet. Das eigentliche soziale Handeln der Beteiligten, und hier insbesondere deren nicht formell geregelte Aktivitäten, bleiben ausgeblendet. Die methodische Unterstützung im Prozeß wird vom Ergebnis her gedacht und ist daher in doppelter Weise verkürzt: a) Die Hilfsmittel sind lediglich an der Erbringung eines Teilergebnisses (nämlich des Software-Produkts) orientiert und nicht an der Veränderung des Arbeitsprozesses als eigentlichem Projektziel. b) Die methodische Unterstützung orientiert sich

nicht am realen Entwicklungsprozeß - der im Kern ein komplexer sozialer Prozeß ist -, sondern an der formal-logischen Abfolge der Entwicklungsschritte zur Entwicklung eines Software-Produkts.

Eine weitergehende Auffassung des Wesens von Software-Projekten findet sich bei Reisin. Sie sieht einen Doppelprozeß von Gestaltung und Modellierung des Produktes einerseits und Kommunikations- und Lernprozessen andererseits. Ihre Überlegungen dienen uns als Grundlage für die Darstellung der Kooperationsbeziehungen im Software-Projekt.[164]

2 Software-Entwicklung als gemeinsamer Lernprozeß

Die Fähigkeit von Menschen, »Neues« zu lernen, wurzelt in ihrem Eingebundensein in ihre konkrete Umgebung. Lernen erfolgt immer in Auseinandersetzung mit den bereits vorhandenen Erfahrungen und dem Wissen, das in einer bestimmten Umwelt gebildet worden ist. Entwickler und Anwender bringen daher in die gemeinsame Lernbeziehung eines Software-Projekts aufgrund ihrer unterschiedlichen Erfahrungshintergründe jeweils spezifische Kompetenzen ein. Die Anforderung an die Lernbeziehung zwischen den Vertretern dieser beiden Gruppen besteht daher darin, diese unterschiedlichen Kompetenzen erfolgreich aufeinander zu beziehen und ausgehend davon zu neuem, gemeinsam gebildeten Wissen zur Gestaltung des Arbeitsprozesses zu gelangen.

Die Stimulierung und die Moderation dieser Lernbeziehung im Software-Projekt ist Teil der Verantwortlichkeit von Entwicklern. Um diese Aufgabe im Software-Projekt erfolgreich bewältigen zu können, benötigen die Entwickler

---

[164] "Wird bei der kooperativen Gestaltung allein das Produkt focussiert, so genügt es im Prinzip, wenn es den Beteiligten gelingt, ihre Kommunikation und ihre wechselseitigen Arbeitstätigkeiten so zu organisieren, daß die verfügbaren individuellen Expertisen und Fertigkeiten im Hinblick auf eine optimale Realisierung des Softwareobjekts eingebracht werden. Ob und in welcher Weise die Individuen ihr Wissen dabei erweitert und verändert haben, interessiert hierbei nicht oder nur am Rande." Und weiter: "Wird jedoch Theoriebildung als Lernprozeß begriffen, so kommt es gerade darauf an, individuelle Veränderungen hervorzurufen. Der Gestaltungsprozeß muß so organisiert werden, daß die an der Theoriebildung Beteiligten sich wechselseitig dazu befähigen, ihre Kenntnisse und Fähigkeiten zu verändern und zu erweitern. Dabei interessiert der sichtbare Entwicklungsfortschritt des gemeinsamen Produkts." [Reisin 1992, S. 132] Sie schlußfolgert daraus hinsichtlich des benötigten Theorie- und Methodenrahmens: "Benötigt wird ein Theorie- und Methodenrahmen, der den Unterschied aufweist zwischen methodischen Mitteln, durch die einerseits der Fortschritt bei der Gestaltung und Modellierung des Produkts unterstützt wird, und solchen, die die Entfaltung von Kommunikations- und Lernprozessen fördern, in denen die Gestaltungskompetenz ausgebildet werden kann." [Reisin 1992, S. 132f.]

- ein Verständnis von den Lernvoraussetzungen, wie sie von Entwicklern und Anwendern in das Projekt eingebracht werden,
- ein Verständnis der Lernerfordernisse, die im Laufe der Software-Entwicklung bewältigt werden müssen,
- die Kompetenz zur Initiierung und Unterstützung der gemeinsamen Lernbeziehung im Projekt.

2.1 Lernbeziehung zwischen Entwickler und Anwender

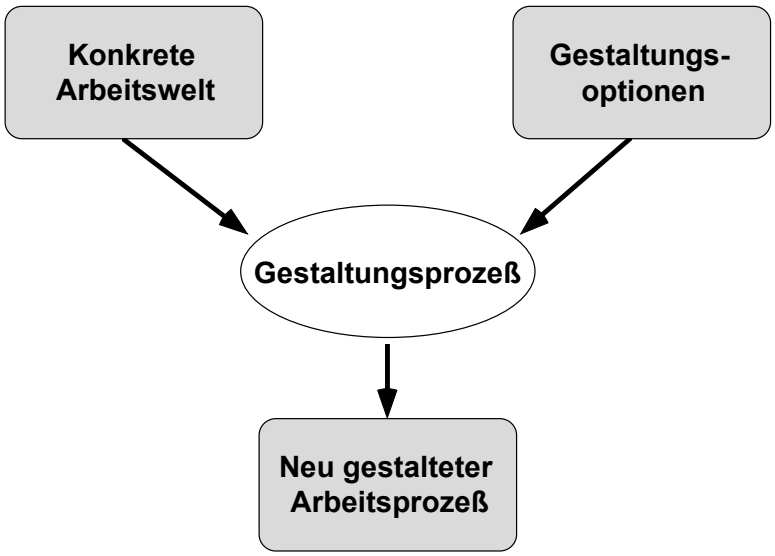
Finn Kensing und Andreas Munk-Madsen [Kensing/Munk-Madsen 1993)[165] schlagen vor, Software-Entwicklung als »Brückenbauen« zu verstehen, weil von zwei unterschiedlichen Bereichen aus etwas gemeinsames Neues erstellt wird. [Kensing/Munk-Madsen 1993, S. 79] Die Software-Entwicklung basiert demnach auf zwei Bereichen: der **konkreten Arbeitswelt** der Anwender und den **Gestaltungsoptionen**, wie sie sich aus den technologischen Möglichkeiten von Hardware und Software sowie den Optionen zur Neugestaltung der Arbeit ergeben. Aus dem Aufeinanderzubewegen beider Bereiche entsteht im Software-Projekt etwas drittes, neues: Ein **neu gestalteter Arbeitsprozeß** mit einem veränderten Computersystem. Die drei wesentlichen Erfahrungsbereiche der Software-Entwicklung sind demnach:

- die konkrete Arbeitswelt.
- die organisatorischen und techologischen Gestaltungsoptionen.
- der neu zu gestaltende Arbeitsprozeß.

---

[165] Die folgenden Überlegungen basieren auf dem Aufsatz von Kensing/Munk-Madsen. Auch wenn wir häufig andere Begriffe verwenden oder auf andere Aspekte hin argumentieren, ist die eigentliche Idee unserer Ausführungen dennoch in dieser Arbeit zu suchen.

Erfahrungsbereiche im Software-Projekt



Quelle: Kensing/Munk-Madsen [1993]

Auf diesen drei Ebenen muß seitens der Projektbeteiligten Wissen aktualisiert oder neu gebildet werden. Dabei ist davon auszugehen, daß Anwender über Erfahrungen und Wissen über die konkrete Arbeit verfügen, während Entwickler Kenntnisse und Erfahrungen im Bereich der (technologischen) Gestaltungsoptionen mitbringen. Die jeweiligen Kompetenzen von Entwicklern und Anwendern in den genannten Erfahrungsbereichen müssen im Entwicklungsprozeß miteinander verbunden werden, um ausgehend davon in einem gemeinsamen Lernprozeß notwendiges Wissen zur Gestaltung des neuen Arbeitsprozesses aufzubauen.

Die Kompetenzen in den drei Erfahrungsbereichen sind weiter zu differenzieren. In einem Projekt wird für jeden der genannten Bereiche einerseits **abstraktes Wissen** gebraucht, um bei der Gestaltung Wesentliches von Unwesentlichem unterscheiden zu können. Andererseits werden **konkrete Erfahrungen** benötigt, um abstraktes Wissen verstehen und überprüfen zu können. Zusammen ergeben sich daraus sechs **Wissensbereiche**, die in einem Software-Projekt mit jeweils unterschiedlichen Voraussetzungen bearbeitet werden.

Wissensbereiche im Software-Projekt

	Konkrete Arbeitswelt	**Neu gestalteter Arbeitsprozeß**	**Optionen**
Abstraktes Wissen	Abstrakte Beschreibung des Arbeitsprozesses (2)	Theorie des zukünftigen Arbeitsprozesses (5)	Zweckgerichtete Beschreibung der technischen und organisatorischen Optionen (4)
Konkrete Erfahrung	Konkrete Erfahrungen mit der betreffenden Arbeitswelt (1)	Konkrete Erfahrungen mit Simulationen des zukünftigen Arbeitsprozesses (6)	Konkrete Erfahrungen mit technischen und organisatorischen Möglichkeiten (3)

Quelle: in Anlehnung an Kensing/Munk-Madsen [1993]

2.1.1 Wissensbereiche im Software-Projekt

Entwickler und Anwender im Software-Projekt haben jeweils unterschiedliche Beziehungen zu den genannten Wissensbereichen. In einem Bereich verfügen sie zu Beginn eines Projekts über viel Vorwissen, das sie im Laufe des Entwicklungsprozesses aktualisieren und den anderen Projektbeteiligten mitteilen müssen, während sie sich das Wissen in den anderen Bereich erst aneignen müssen. Generell gilt: Alle Beteiligten - Entwickler wie Anwender - brauchen ein gewisses Maß an Erfahrungen und Wissen über alle genannten Bereiche, um ihre Rolle im Entwicklungsprozeß erfolgreich einnehmen zu können. Der Lernprozeß im Software-Projekt ist daher so zu gestalten, daß dieses erforderliche Gestaltungswissen gemeinsam aufgebaut werden kann.

Konkrete Erfahrungen mit der betreffenden Arbeitswelt (Bereich 1)

Das Ziel des Software-Projekts ist die Veränderung der konkreten Arbeitswelt. Die Arbeitswelt muß daher der Ausgangspunkt und der zentrale Bezugspunkt der Lernprozesse im Projekt sein. Organigramme oder Stellenbeschreibung, als Hilfsmittel herangezogen, vermitteln nur einen ersten Eindruck dieser Realität, sind aber keine ausreichend genaue Darstellung. Die Wirklichkeit läßt sich nur über die Erfahrung der dort arbeitenden Menschen erschließen.

Über diese Erfahrungen verfügen Anwender. Sie beziehen sich auf die verwendeten Arbeitsmittel und -gegenstände sowie deren Gebrauch, auf die erforderlichen Kooperationsbeziehungen, die umgebende Unternehmensumwelt und die mit der Arbeit verfolgten konkreten Ziele und Zwecke.

Um diese Arbeit im Software-Projekt gestalten zu können, benötigen Entwickler einen Zugang zu diesen Erfahrungen. Sie müssen diese Arbeit verstehen.

Abstrakte Beschreibung des Arbeitsprozesses (Bereich 2)

Mit dem Verstehen der Arbeit erzeugen Anwender und Entwickler Schritt für Schritt eine abstrakte Beschreibung des Arbeitsprozesses, die in verallgemeinerter Form alle wesentlichen gestaltungsrelevanten Aspekte enthält. Diese Beschreibungen stellen ein Bezugsschema dar, das im Entwicklungsprozeß für die Identifikation von Gestaltungswünschen sowie als Evaluationsgrundlage für die Auswirkungen angestrebter Veränderungen dient.

Weder Entwickler noch Anwender verfügen zu Beginn des Projekts über dieses abstrakte Wissen. Sie müssen sich diese Verallgemeinerungen im Laufe des Prozesses gemeinsam erarbeiten[166], ein gemeinsames »Bild« von der konkreten Arbeitswelt entwickeln und den "Realitätsbereich" des Software-Projekts erschließen. [Reisin 1992)[167]

---

[166] Das klassische Software-Engineering hält für diesen Abstraktionsprozeß eine Reihe von Methoden der Anforderungsanalyse bereit. Diese Beschreibungsmethoden bauen darauf auf, daß die Entwickler das erforderliche Wissen von den Anwendern erfragen und in Form von Modellen abtragen. Diese Form der Abstraktion hat nach unserer Aufassung zwei wesentliche Nachteile: Als Methode zur Abstraktion sind sie für die Anwender ungeeignet, weil sie viel Erfahrung und entsprechende Kenntnis erfordern, damit ein kompetenter Umgang mit ihnen möglich ist. Da Entwickler meist über diese Kenntnisse verfügen, während Anwender nicht gewohnt sind, mit diesen Beschreibungsmethoden zu Modellen über ihre konkrete Arbeitssituation zu gelangen, entwickelt sich zwischen den beiden Gruppen ein Ungleichgewicht. Letztendlich wird dann in diesen Modellen lediglich das festgehalten, was den Entwicklern verallgemeinerungswert erschien, während Anwendern - geblendet von der Scheinplausibilität der Rauten und Ellipsen - die Kompetenz zu einer fundierten Teilnahme am Abstraktionsprozeß fehlt. Ein weiteres Problem des Einsatzes dieser Beschreibungsmethoden zur Anforderungsanalyse besteht darin, daß gerade die sozialen und subjektiven Momente der Arbeitssituation durch die vermeintliche »Universalität« der Beschreibungsmethoden »wegabstrahiert« werden. [Reisin 1992, S. 121]

[167] In Abgrenzung zur klassischen Auffassung des Software-Engineering, wonach die Entwicklung einer abstrakten Beschreibung des Arbeitssystems als »Abbildung der Realität« oder als »Ermittlung eines Ist-Zustands« diene, verweist Reisin auf den Aspekt der bewußten Erschließung eines gemeinsamen Realitätsbereichs: "Entgegen den vorherrschenden Annahmen wird in diesen Prozessen nicht ein Ist-Zustand ermittelt, sondern von Anfang an ein bereits veränderter Gegenstandsbereich realabstraktiv exploriert und antizipatorisch neu gestaltet." [Reisin 1992, S. 174]

Konkrete Erfahrungen mit Gestaltungsmöglichkeiten (Bereich 3)

Die Gestaltungsoptionen im technischen und organisatorischen Bereich sind für die Projektbeteiligten zentraler Bezugspunkt der gemeinsamen Entwicklungsarbeit. Das Wissen und die Erfahrungen darüber, was möglich ist, bieten die Grundlage für das Entwickeln eigener Visionen und Wünsche.

Entwickler verfügen von Berufs wegen über Erfahrungen hinsichtlich der Gestaltungsmöglichkeiten. Diese beziehen sich aber vorrangig auf die Optionen zur technischen Gestaltung, während organisatorische Gestaltungsmöglichkeiten weniger zum bewußten Erfahrungsbereich gehören. Entwickler benötigen zur erfolgreichen Bewältigung des Software-Projekts konkrete Erfahrungen hinsichtlich der organisatorischen Folgewirkungen entwickelter technischer Lösungen. Diese sind im Entwicklungsprozeß erforderlich, um die konkreten Erfahrungen und Verallgemeinerungen der Anwender erfassen und die Wirkung eigener Gestaltungsvorschläge einschätzen zu können.

Anwender benötigen konkrete Vorstellungen von den technologischen und organisatorischen Gestaltungsmöglichkeiten, um einen Bezugspunkt für die Entwicklung von Wünschen und Vorstellungen zur Veränderung ihrer konkreten Arbeitssituation zu entwickeln. Erfahrungen über Gestaltungsmöglichkeiten sind für sie weiterhin notwendig, um die Folgewirkungen bestimmter technisch-organisatorischer Veränderungen abschätzen zu können.

Zweckgerichtete Beschreibung der technischen und organisatorischen Optionen (Bereich 4)

Ein Überblick über die für diesen Fall in Frage kommenden technischen und organisatorischen Gestaltungsoptionen ist für den Erfolg des Projekts erforderlich, um Gestaltungsideen und -visionen entwickeln zu können und die Möglichkeiten und Grenzen von entwickelten Gestaltungsvorschlägen einschätzen zu können.

Entwickler benötigen einen Überblick darüber, welche der auf dem Markt befindlichen Hard- und Softwareangebote für den konkreten Fall verwendbar sind, um Anregungen für Gestaltungen geben zu können und die Grenzen gewählter oder vorgesehener technischer Lösungen einschätzen zu können.

Während Wissen hinsichtlich technischer Angebote über die Entwickler oft in das Projekt hineingebracht werden kann, fehlt das Wissen über organisatorische Gestaltungsmöglichkeiten in Software-Projekten meist vollständig. In dem fehlenden Überblick über organisatorische Gestaltungsoptionen liegt oft der Mangel an Visionen zur Neugestaltung des Arbeitsprozesses begründet. Das fördert die einseitige Konzentration auf die technische Gestaltung und behindert die Suche nach neuen Formen menschengerechter Arbeitsgestaltung. Wenn die-

ses Wissen nicht im Unternehmen mobilisiert werden kann, bietet es sich an - insbesondere bei Projekten, die tiefgreifende organisatorische Veränderungen intendieren - externe Berater wie Arbeitswissenschaftler, -soziologen oder -psychologen fallweise oder generell hinzuzuziehen.

Theorie des zukünftigen Arbeitsprozesses (Bereich 5)

Das inhaltliche Zentrum der Neugestaltung eines Arbeitsprozesses ist eine gemeinsam entwickelte Theorie[168]. In dieser Theorie werden die wesentlichen Aspekte des zukünftigen Arbeitsprozesses in allgemeiner Form beschrieben. Sie beinhaltet eine Antizipation der zukünftigen Tätigkeiten, der zu gestaltenden Arbeitsmittel und -gegenstände sowie der angestrebten Kooperationsbeziehungen.

Die Theorie ist für Anwender und Entwickler gleichermaßen die gültige inhaltliche Beschreibung des angestrebten Arbeitsprozesses. Für die Entwickler enthält sie darüber hinaus die Anforderungen an die zu erstellenden Software-Produkte.

Konkrete Erfahrungen mit Simulationen des zukünftigen Arbeitsprozesses (Bereich 6)

Konkrete Erfahrungen mit Simulationen des künftigen Arbeitsprozesses werden im Laufe des Gestaltungsprozesses als Orientierungspunkt für die Qualität der entwickelten Gestaltungsvorschläge benötigt.

Anwender brauchen entsprechende konkrete Erfahrungen, um allgemeine Gestaltungsvorschläge auf ihre konkrete Arbeitssituation beziehen zu können und sie darauf aufbauend besser einschätzen zu können. Weiterhin benötigen sie diese Erfahrungen, um die Qualität entwickelter Gestaltungslösungen im konkreten Gebrauch beurteilen zu können.

---

[168] In der Tradition der "theory building view" (Naur) wird die Erstellung von Software vorrangig als **Theoriebildungsprozeß** angesehen. Die Erstellung des Produkts, die bei klassischen Betrachtungsweisen des Software-Engineering im Vordergrund steht, ist diesem Prozeß der Entwicklung einer Theorie über den zu gestaltenden Realitätsbereich logisch nachgeordnet. [Naur 1985] Reisin greift diese Überlegungen auf und charakterisiert den Software-Entwicklungsprozeß auf Holzkamp und dessen Begründung der zentralen Bedeutung der "Zweckantizipation" bei der Werkzeugherstellung bezugnehmend wie folgt: "Demnach ist Softwareherstellung immanent als kreativer Arbeitsprozeß zu bestimmen, in dem neue ideelle Bezugsschemata aufgebaut werden müssen, bevor sie im Softwareobjekt vergegenständlicht werden können. Den Prozeß des Aufbaus einer neuen Zweckantizipation verstehe ich als Theoriebildung über die Gebrauchszwecke der künftigen Softwareobjekte." [Reisin 1992, S. 93f.] Dabei verdeutlicht sie, daß die Theoriebildung als **gemeinsamer Lernprozeß** von Anwendern und Entwicklern zu betrachten ist.

Entwickler benötigen konkrete Erfahrungen mit Simulationen des künftigen Arbeitsprozesses, um die Übereinstimmung von angestrebten und realen Veränderungen zu überprüfen und damit die Gestaltungsvorschläge zu evaluieren.

2.1.2 Entwickler und Anwender im Verhältnis zu den Wissensbereichen

Keine der beiden im Software-Projekt beteiligten Gruppen verfügt über ausreichendes Wissen in allen sechs Bereichen. Entwickler und Anwender verfügen über Wissen in einem oder zwei Bereichen, während sie sich das erforderliche Wissen in den anderen Bereichen zunächst aneignen müssen. Das erforderliche abstrakte Wissen und die notwendigen konkreten Erfahrungen bezüglich der Neugestaltung des Arbeitsprozesses sind darüber hinaus weder bei den Anwendern noch bei den Entwicklern zu Beginn eines Projekts vorhanden. [Reisin 1993]

Zu fragen ist also:

a) Über welches Wissen aus welchen Wissensbereichen verfügen Entwickler und Anwender jeweils zu Beginn eines Projekts?

b) Welches Wissen aus welchen Wissensbereichen müssen sie sich im Laufe des Projekts erst aneignen?

Wissen und Lernerfordernisse der Anwender

Zu Beginn eines Projekts verfügen Anwender über konkrete Erfahrungen mit ihrer Arbeitswelt. Diese Erfahrungen beziehen sich auf die verwendeten Arbeitsmittel und -gegenstände und deren Gebrauch, auf die erforderlichen Kooperationsbeziehungen, die umgebende Unternehmensumwelt und die mit der Arbeit verfolgten konkreten Ziele und Zwecke. Sie beziehen sich aber auch auf die sozialen und subjektiven Aspekte der Arbeit, die für ein konkretes Arbeitssystem konstitutiv sind.

Diese Erfahrungen sind nur zu einem kleinen Teil im Sinne von regelhaftem Wissen formulierbar. Der Großteil des im Arbeitsprozeß erworbenen Wissens ist "Erfahrungswissen" [Malsch 1987], über das Menschen nicht in Form von Regeln oder klar umrissenen Begriffen verfügen um es einfach an andere »weiterzureichen«.[169] Dieser Umstand verdient bei der Bearbeitung des Koopera-

[169] Gegenüber einer Aneignung durch Dritte, wie sie in klassischen Anforderungsanalysen intendiert ist, verhält sich diese Art des Wissens »sperrig«. Brödner sieht hierin eine wesentliche Grundlage für das Scheitern vieler Vernetzungsprojekte im Fertigungsbereich. [Brödner 1985]

tionsprozesses im Projekt als Lernprozeß besondere Beachtung und wird an anderer Stelle genauer diskutiert.

Anwender verfügen nur in Ausnahmefällen über einen Überblick auf technische und organisatorische Gestaltungsmöglichkeiten. Noch viel weniger ist vorauszusetzen, daß sie über konkrete Gestaltungserfahrungen verfügen. Auch ein abstraktes Wissen über ihre konkrete Arbeitswelt, wie es für die Gestaltung benötigt wird, ist keineswegs selbstverständlich. Für das Bestehen im Arbeitsalltag reichen die konkreten Erfahrungen der Anwender vollkommen. Abstraktes Wissen, wie es für die Software-Entwicklung erforderlich ist, ist hierfür kaum notwendig. Ganz anders bei der Neugestaltung des Arbeitsprozesses; hier ist es wichtig, über verallgemeinerbare und gestaltungsrelevante Aussagen zu verfügen. Notwendig ist also, daß Anwender aus ihren konkreten Erfahrungen abstraktes Wissen bilden, also etwas »neues« lernen. Dabei ist davon auszugehen, daß die Anwender diese Lernprozesse nicht aus sich selbst heraus vollziehen, sondern nur in Austauschprozessen mit den Entwicklern.

Die erfolgreiche Bewältigung eines Software-Projektes macht es erforderlich, daß Anwender im Verlauf eines Entwicklungsprozesses Kompetenzen bezüglich folgender Bereiche aufbauen müssen:

- eine abstrakte Vorstellung über ihre Arbeitswelt (Bereich 2),
- eine allgemeine Theorie des zukünftigen Arbeitsprozesses (Bereich 5),
- konkrete Erfahrungen mit Simulationen des künftigen Arbeitsprozesses (Bereich 6).

Für eine erfolgreiche Bewältigung ihrer Rolle im Entwicklungsprozeß müssen sich Anwender eine abstrakte Vorstellung über ihre Arbeitswelt (Bereich 2) und eine allgemeine Theorie des zukünftigen Arbeitsprozesses (Bereich 5) erarbeiten. Die Kompetenz in beiden Bereichen dient dazu, Gestaltungsanforderungen definieren zu können und vorgeschlagene Lösungen überprüfen zu können. Die Theoriebildung wird unterstützt durch konkrete Erfahrungen mit dem durch Prototypen simulierten Arbeitssystem, das wesentliche Gestaltungsmerkmale des zukünftigen Arbeitsprozesses besitzt (Bereich 6). Bei modernen Software-Projekten gelingt es den Anwendern nur über den konkreten Gebrauch zu modellhaften Vorstellungen für die Neugestaltung zu gelangen. Die Vorschläge zur Unterstützung des Software-Entwicklungsprozesses durch den Einsatz von Prototypen basieren unter anderem auf diesem Lernverhalten von Anwendern. [Holz auf der Heide 1993, Budde u.a. 1992]

Wissen und Lernerfordernisse der Entwickler

Zu Beginn eines Projektes verfügen Entwickler über abstraktes Wissen und konkrete Erfahrungen hinsichtlich der technischen Gestaltungsoptionen. Das ergibt sich aus ihrer Berufserfahrung und ihrem originären fachlichen Können. Kenntnisse oder Erfahrungen hinsichtlich der Rückwirkungen technischer Gestaltungsmaßnahmen auf die Arbeit oder organisatorische Veränderungen fehlen ihnen in aller Regel.[170]

Entwickler verfügen nur in seltenen Fällen über allgemeine Erfahrungen mit dem betreffenden Arbeitsprozeß. Wenn in ähnlichen Projekten bereits Erfahrungen gesammelt wurden oder durch die Zugehörigkeit zu dem Unternehmen, in dem das Projekt stattfindet, bereits Kenntnisse vorliegen, erleichtert dies die Verständigung mit den Anwendern, ersetzt sie aber keineswegs. Wesentliche Aspekte eines Arbeitsprozesses sind subjektiver oder sozialer Natur und nur über das Erfahrungswissen der Anwender zu erschließen.

Entwickler müssen im Laufe des Projekts Wissen in folgenden Bereichen neu bilden:

- eine abstrakte Vorstellung über ihre Arbeitswelt (Bereich 2),
- eine allgemeine Theorie des zukünftigen Arbeitsprozesses (Bereich 5),
- konkrete Erfahrungen mit Simulationen des künftigen Arbeitsprozesses (Bereich 6).

Entwickler müssen sich eine abstrakte Vorstellung des Arbeitsprozesses (Bereich 2) und eine allgemeine Theorie des zukünftigen Arbeitsprozesses (Bereich 5) im Projektverlauf erarbeiten, weil diese die Grundlage für die Neugestaltung des Arbeitsprozesses und für die technische Realisierung des Software-Produkts sind. Für den Test entwickelter Gestaltungsvorschläge benötigen sie konkrete Erfahrungen mit Simulationen des künftigen Arbeitsprozesses (Bereich 6). Die Lernprozesse in den genannten Bereichen werden in aktiver Auseinandersetzung mit den Anwendern im Projekt vollzogen. Sie korrespondie-

---

[170] Durch die Beschränkung von Entwicklern auf die technische Seite des Gestaltungsprozesses werden in der praktischen Arbeit beständig anfallende Möglichkeiten zum Sammeln von Erfahrungen nicht genutzt. Wenn diese Ereignisse von den Entwicklern nicht bewußt verarbeitet werden, fehlen ihnen die kognitiven Strukturen, um sie als Erfahrungen zu »speichern«; ein gutes Beispiel dafür, in welcher Weise die Kompetenz von Entwicklern durch die berufliche Identität beeinflußt wird. Weil Entwickler meist nur eine eingeschränkte Verantwortung für die technische Seite des Entwicklungsprozesses übernehmen und keine Verantwortlichkeit für den ganzheitlichen Prozeß, werden beständig erfahrene Ereignisse nicht zu Erfahrungen verarbeitet.

ren mit Lernprozessen, die die Entwicklergruppe zumeist für sich organisiert. Hierbei werden vorhandene Erfahrungen über die (technischen) Gestaltungsoptionen (Bereich 3) auf das zur Debatte stehende Gestaltungsproblem bezogen und allgemeines Wissen über technologische (und organisatorische) Optionen aufgebaut (Bereich 4).

2.2 Zusammenfassung

Entwickler und Anwender bringen in die gemeinsame Lernbeziehung eines Software-Projekts aufgrund ihrer unterschiedlichen Erfahrungshintergründe jeweils spezifische Kompetenzen ein. Das Problem der Lernbeziehung zwischen den Vertretern dieser beiden Gruppen besteht darin, die unterschiedlichen Kompetenzen erfolgreich aufeinander zu beziehen und ausgehend davon zu neuem, gemeinsam gebildetem Wissen zur Gestaltung des Arbeitsprozesses zu gelangen. Die Stimulierung und die Moderation dieser Lernbeziehung im Software-Projekt ist Teil der Verantwortlichkeit von Entwicklern.

3 Konflikt und Konsens in der Software-Entwicklung

Der Erfolg von Software-Projekten ist in vielfacher Hinsicht damit verbunden, inwieweit es gelingt, die mit dem Projekt verknüpften Konflikte zu bewältigen. Die Auslöser dieser Konflikte sind vielfältiger Natur: sie resultieren z.B. aus unterschiedlichen Sichtweisen verschiedener Projektbeteiligter oder aus den Veränderungen der Arbeitsbedingungen einzelner Mitarbeiter oder sogar ganzer Abteilungen. Versucht man, den Ursachen dieser Konflikte auf die Spur zu kommen, so stellt man fest, daß diese nicht in besonderen »menschlichen Schwächen« wurzeln, sondern vor allem in der **betriebspolitischen Bedeutung** von Software-Projekten.

Software-Projekte greifen tief in das betriebliche Gleichgewicht von Einfluß und Kontrolle ein und verändern die Arbeitsbedingungen der Beschäftigten grundlegend. Da die Entwicklung von Software auf die Neugestaltung eines Arbeitsprozesses zielt, ist die betriebspolitische Verarbeitung der damit einhergehenden Konflikte untrennbar mit dem Fortgang des Projekts verbunden. Software-Projekte können nicht jenseits dieser Auseinandersetzungen zum Erfolg gebracht werden, sondern nur durch einen konstruktiven Umgang mit ihnen.

Interessenauseinandersetzungen wirken als Umfeldfaktoren in vielfältiger Weise auf Software-Projekte ein. Auseinandersetzungen im Projektumfeld können sich so schwerwiegend auswirken, daß die Realisierung vollständig gefährdet wird. Ortmann u.a. berichten beispielsweise von einem Projekt, das durch Auseinandersetzungen zwischen zwei Abteilungen über Jahre blockiert wurde. [Ortmann

u.a. 1990] Weltz und Ortmann verweisen anhand einer empirischen Untersuchung darauf, daß die Lösung der betriebspolitischen Konflikte ein erstrangiger Erfolgsfaktor von Software-Projekten ist. [Weltz/Ortmann 1992]

Die betriebspolitischen Konflikte wirken direkt auf die Binnenstrukturen der Projekte. Jeder ist an diesem Projekt mit Interessen und Vorstellungen beteiligt und ist daher auch immer Partei. Diese »Parteilichkeit« kann die sozialen Beziehungen in Projekten negativ beeinflussen und zu Projekt-Krisen bis hin zum faktischen Scheitern führen. Dies ist insbesondere dann der Fall, wenn die Konflikte und Interessensgegensätze der Akteure nicht offen thematisiert und damit einem rationalen Diskurs entzogen werden. Gerade das Hochhalten vermeintlich »objektiver« technischer Kriterien ist oft nur ein deutlicher Ausdruck dafür, daß die Parteien mit besonderer Vehemenz gewillt sind, die dahinterliegenden Probleme nicht zu thematisieren. Diese Einstellung führt in Projekten dazu, daß der notwendigen Abstimmung der Parteien und dem Austragen von Konflikten kein ausreichender Raum geboten wird. Statt dessen wird so getan, als seien lediglich technische Fragen nach zweckrationalen Kriterien zu bearbeiten. Gerade in der Tatsache, daß Projekte keinen Raum bieten, um Interessenauseinandersetzungen auszutragen, liegen die Wurzeln vieler Projekt-Krisen. [ebd., S. 135]

Gleichzeitig sind die Interessen und Motive der Projektbeteiligten der eigentliche Motor der sozialen Prozesse im Projekt. Software-Projekte sind von Erfolgsfaktoren bestimmt, die ein Handeln der Projektbeteiligten erfordern, welches sich nicht anordnen läßt. Das »Sich-Verstehen-Wollen« in Kommunikationsprozessen, die Bereitschaft, Auseinandersetzungen offen miteinander auszutragen oder gemeinsam lernen zu wollen, sind Anforderungen, die durch keine Arbeitsanweisung erzwungen werden können. Sie bedürfen der Motivation der Projektbeteiligten, die eng mit ihren Vorstellungen und Interessen verbunden ist. Die Suche nach gemeinsamen Konfliktlösungen verschafft den sozialen Prozessen im Software-Projekt ihre Dynamik.

3.1 Konfliktstrukturen von Software-Projekten

Für das Software-Projekt sind zunächst die Konflikte von Bedeutung, die im Projektumfeld angelagert sind und oft »hinter den Kulissen« auf das Projekt einwirken. Diese nennen wir im folgenden **»projekt-externe Konflikte«**. Sie finden nicht unmittelbar im Projekt, d.h. zwischen einzelnen Projektbeteiligten statt, sondern zwischen einzelnen Personen oder organisatorischen Einheiten des Unternehmens. Sie können zu Konflikten zwischen einzelnen Projektbeteiligten werden, wenn mindestens zwei Teilnehmer des Projekts jeweils unterschiedlich Partei ergreifen. Dabei gelten nur diejenigen Auseinandersetzungen als extern,

auf die das Projekt und die Projektleitung keinen direkten, über organisatorische Vereinbarungen geregelten Zugriff haben. Sofern direkter Einfluß des Projekts besteht, sehen wir Konflikte als »projekt-interne Konflikte« an.

Projekt-externe Konflikte entstehen durch das Eingreifen des Software-Projekts in das Verhältnis zwischen einzelnen Beschäftigten oder ganzen organisatorischen Einheiten. Hierbei handelt es sich oft um Auseinandersetzungen, die dadurch hervorgerufen werden, daß Aufgaben oder Einflußbereiche durch die Neugestaltung des Arbeitsprozesses von einem in einen anderen Bereich verlagert oder anderweitig berührt werden:

- Abteilungen haben plötzlich keinen Zugriff mehr auf bestimmte Informationen, weil die Bearbeitung der entsprechenden Vorgänge auf einen anderen Bereich übergehen soll.
- Vorgesetzte haben beispielsweise durch die Übertragung eines bisher über Ordner verwalteten Datenbestandes auf Computer weniger Kontrolle über den Arbeitsstand der Mitarbeiter.
- Arbeitsplätze in bestimmten Unternehmensbereichen sollen abgebaut werden, weil die hier anfallenden Aufgaben durch den Einsatz des Computers von weniger Beschäftigten erledigt werden können.

In all diesen Fällen entstehen Konflikte, die als wichtige Umfeldbedingungen der Projekte beachtet werden müssen. Die Weigerung, dem Projektteam Informationen zur Verfügung zu stellen, können ebenso aus dieser betriebspolitischen Verquickung von Software-Projekten resultieren, wie plötzliches »Sperrfeuer« einflußreicher Personen, die Festlegung des Projekts auf sachlich falsche Vorgaben oder ein Veto des Betriebsrates. Wer als Entwickler nicht weiß, daß diese Konfliktstrukturen vorliegen, wundert sich nur allzuoft über unzureichende Informationen, plötzliche Mittelkürzungen, unsachgemäße Vorgaben oder die Aussetzung des Projekts.

Als »**projekt-interne Konflikte«** bezeichnen wir die Konflikte, die zwischen mindestens zwei der mittelbar oder unmittelbar am Projekt beteiligten Personen oder organisatorischen Einheiten stattfinden, wenn direkter Einfluß auf die Konfliktparteien seitens des Projekts besteht.

Diese Konflikte haben ihre Ursachen darin, daß mit dem Projekt die Interessen der Beteiligten in der einen oder anderen Weise betroffen sind. Sie wurzeln darüber hinaus in unterschiedlichen Sichtweisen und Grundauffassungen der Projektmitglieder. Die empirische Untersuchung von Weltz und Ortmann gibt hier einige Aufschlüsse:

- Konflikte bestanden zwischen Entwicklern und Anwendern. Das Verhältnis beider Gruppen war oft durch sehr unterschiedliche Grundorientierungen bei der Interpretation der Projektaufgabe bestimmt. Entwickler stellten häufig technische Aspekte der Gestaltungsaufgabe in den Vordergrund. Die von ihnen angestrebte Qualität des Produkts bezog sich vorrangig auf technische Qualitätskriterien. Demgegenüber stand für Anwender vorrangig der Aufgaben- und Anwendungsbezug während des Projekts im Vordergrund. Ihre Ziele waren daher eher »Erleichterung bei der Aufgabenerledigung« oder »Belastungsreduzierung«.
- Konflikte bestanden weiterhin zwischen den am Projekt beteiligten Anwendern. Die Differenzen zwischen verschiedenen Anwendern resultierten häufig daraus, daß mit der Gestaltung des Arbeitsprozesses die Arbeitsaufgaben oder das Verhältnis zwischen verschiedenen Abteilungen verändert werden sollte. Gerade diese Interessenauseinandersetzungen wurden oft hinter technischen Anforderungen versteckt.
- Konflikte bestanden weiterhin zwischen den am Projekt beteiligten Entwicklern. Differenzen innerhalb der Entwicklergruppe ergaben sich oft daraus, daß mit bestimmten technischen Entscheidungen konkrete Vor- oder Nachteile verbunden waren. Mit der Wahl einer bestimmten Programmiersprache wurden z.B. die Karriereaussichten und Qualifizierungschancen bestimmter Entwicklergruppen nachhaltig beeinflußt. [Weltz/Ortmann 1992, S. 125ff.]

Die mit der Software-Entwicklung verbundenen Konflikte weisen einen sehr unterschiedlichen **Charakter** auf. In der skandinavischen Diskussion um partizipatives Software Engineering identifiziert Reisin [1993] zwei unterschiedliche Auffassungen der Interpretation der Konfliktstrukturen bei der Software-Entwicklung. Ein Ansatz betrachtet die Auseinandersetzungen in der Software-Entwicklung unter der Perspektive der »Harmonie«, ein anderer betont demgegenüber die »Konfliktperspektive«. Die »Harmonieperspektive« geht davon aus, daß die Auseinandersetzungen bei der Software-Entwicklung wesentlich auf Differenzen und Mißverständnisse zurückzuführen sind, die durch Gespräche und Informationen gelöst werden können. Die sozialen Verhältnisse sind demnach durch eine prinzipielle Harmonie gekennzeichnet. Demgegenüber gehen die Vertreter der »Konfliktperspektive« davon aus, daß die Beziehungen der beteiligten Gruppen grundsätzlich durch unlösbare Konflikte gekennzeichnet sind. Ausgehend von einem arbeitnehmernahen Standpunkt gehen sie davon aus, daß bei der Software-Entwicklung eine Konfliktthaltung zur Durchsetzung der Interessen der Beschäftigten eingenommen werden müsse. Auftretenden Konflikten sei mit

Konfrontation, Verhandlung und Kompromiß zu begegnen. [Nygaard/Sørgaard 1989, S. 388][171]

Betrachtet man den Charakter der Auseinandersetzungsstrukturen in Software-Projekten und deren Umfeld, so lassen sich in der Realität für beide Sichtweisen konkrete Beispiele finden. Einerseits existieren Konflikte, die wesentlich auf unterschiedlichen Sichtweisen oder auch Mißverständnissen der Beteiligten beruhen und im Verlauf eines gemeinsamen Prozesses durch einen Konsens gelöst werden können. In diesen Konflikten weisen die Beteiligten gleiche oder zumindest ähnliche Interessen auf. Ein Konsens ist möglich, ohne daß einer der beiden Beteiligten grundlegende Interessenpositionen aufgeben muß. Wir nennen diese Konflikte künftig **»konsensual lösbare Konflikte«**. Andererseits lassen sich in der Realität aber auch unschwer Auseinandersetzungen identifizieren, die nur die Durchsetzung eines der beiden Standpunkte zulassen oder einen Kompromiß[172] erfordern, bei dem beide Seiten Aspekte ihres Standpunktes aufgeben müssen. In diesen Konflikten besteht zwischen den Positionen der beiden Parteien ein unlösbarer Widerspruch. Diese Art von Konflikten entsteht, wenn das zur Debatte stehende Problem grundlegende Interessenpositionen der Konfliktparteien berührt und nur die Durchsetzung höchstens einer der beiden Interessenspositionen zuläßt. Wir nennen diese Konflikte künftig **»antagonistische Konflikte«**, weil die Durchsetzung einer der beiden Positionen immer nur zu Lasten der jeweils anderen möglich ist.

In der Zusammenfassung der geschilderten Konfliktarten erhalten wir folgende Matrix:

---

[171] Die Vertreter des Forschungsprojekts »PEtS« [Floyd u.a. 1994] und Oppermann [1983] plädieren in beteiligungsorientierten Projekte für eine »Konsensperspektive«. Ausgehend davon, daß die Kooperation im Projekt auf Dauer nicht in der Perspektive grundsätzlicher Konflikte denkbar ist, schlagen sie vor, einen gemeinsamen Konsens aller beteiligten Gruppen zu Beginn eines Projekts zu vereinbaren und diesen als unverzichtbare Bedingung für den Start des Entwicklungsprozesses von partizipativ durchgeführten Projekten aufzufassen. [Reisin 1993; Reisin 1991]

[172] Als Kompromiß bezeichnen wir das Ergebnis eines Konflikts, das dann zustande kommt, wenn keine der beiden Konfliktparteien die andere überzeugen kann oder sich durch den Einsatz von Macht durchsetzen kann. Kompromisse werden meist dann geschlossen, wenn die Standpunkte der Parteien unvereinbar sind und beide Seiten aufeinander angewiesen sind oder es zu sein glauben.

Konfliktstrukturen von Software-Projekten

	konsensual-lösbar	**antagonistisch**
projekt-intern	konsensual-lösbare, interne Konflikte	antagonistische interne Konflikte
projekt-extern	konsensual-lösbare, externe Konflikte	antagonistische externe Konflikte

3.2 Software-Entwicklung im Spannungsfeld antagonistischer Konflikte

Antagonistische Konflikte bestehen dann, wenn von mindestens zwei Parteien bezogen auf das Software-Projekt Konfliktpositionen eingenommen werden, die auf unvereinbaren, grundlegenden Interessen beruhen. Diese Konflikte entstehen dann, wenn die angestrebte Neugestaltung des Arbeitsprozesses nur zu erreichen ist, indem fundamentale Interessen einzelner Beschäftigter oder ganzer organisatorischer Einheiten verletzt werden.

Ein besonders krasses Beispiel für antagonistische Konfliktstrukturen ist es, wenn durch die neu zu entwickelnde Software der Wegfall von Arbeitsplätzen angestrebt wird, ohne daß den Betroffenen andere Arbeitsplätze angeboten werden. Antagonistische Konflikte sind auch dann gegeben, wenn die Neuverteilung von Kompetenzen oder Aufgaben zwischen einzelnen Arbeitsplätzen oder Abteilungen jeweils zu Lasten einer der Parteien geregelt werden soll, ohne daß die jeweils unterliegende Seite einen Ausgleich erhält. In beiden Fällen ist es so, daß die Durchsetzung des einen Standpunkts die Nicht-Durchsetzung des anderen bedingt.

Antagonistische Konfliktstrukturen treten häufig dann auf, wenn mit dem Entwicklungsauftrag des Software-Projekts die Verletzung grundlegender Interessen von einzelnen Beschäftigten oder ganzer Beschäftigtengruppen verbunden ist. Dies geschieht in deutlicher Weise im Falle des Arbeitsplatzabbaus. Aber auch unterhalb dieser Schwelle kann die angestrebte Neugestaltung zu grundlegenden Verschlechterungen für die Mitarbeiter führen. Diese liegen dann vor, wenn die Interessen der Beschäftigten an einer menschengerechten Gestaltung ihrer Arbeit verletzt werden. Als allgemeine Gestaltungskriterien humaner Arbeitsgestaltung gelten nach Hacker:

- Ausführbarkeit
  Die Arbeit muß mit den zur Verfügung stehenden Werkzeugen sowie den Fähigkeiten des Menschen ausführbar sein.
- Schädigungslosigkeit
  Arbeit soll so gestaltet sein, daß sie nicht zu einer psychischen oder physischen Schädigung der Gesundheit der Beschäftigten führt.
- Beeinträchtigungslosigkeit
  Arbeit soll so gestaltet sein, daß das körperliche oder psychische Wohlbefinden der Beschäftigten nicht beeinträchtigt ist.
- Persönlichkeitsförderlichkeit
  Arbeit soll so gestaltet sein, daß die Persönlichkeitsentwicklung im Sinne der Entfaltung der Potentiale und Kompetenzen gewährleistet ist. [Hacker 1986]

Die Gefahren, die in der Verletzung fundamentaler Interessen der Beschäftigten für den Projekterfolg liegen, beschreibt Reisin wie folgt: "Kooperation und Kreativität können nicht angeordnet werden. Vor dem Hintergrund bestehender Herrschaftsverhältnisse und Machtstrukturen in Einsatzorganisationen birgt jede Veränderung des status quo aus Arbeitnehmersicht potentiell die Verschlechterung der eigenen Arbeitssituation. Eine nur auf die Bereitstellung ihres Fachwissens reduzierte Zusammenarbeit mit den »Agenten der Veränderung«, den Software-Entwicklern, wird von den Benutzern als »Wissensaneignung« aufgefaßt, die zudem die berufliche Existenz gefährde." [Reisin 1993)

Wenn ein Software-Projekt im Spannungsfeld antagonistischer Interessen angesiedelt ist, ist der Projekterfolg in den meisten Fällen stark gefährdet. Betroffene Abteilungen verweigern die Zusammenarbeit, Entwickler erhalten fehlerhafte Informationen und Projektbeteiligte werden gegeneinander ausgespielt. Bei projekt-externen Konflikten, die sich per definitionem dem unmittelbaren Zugriff der Projekte entziehen, kann seitens der Projektbeteiligten nur darauf gedrängt werden, daß diese Konflikte offen zutage treten. Solange sie als »schwelende Konflikte« existieren, können sie unabsehbare Folgen für den Projekterfolg haben.

Werden die Auseinandersetzungen projekt-intern ausgetragen, besteht zwar im Grunde eine Eingriffsmöglichkeit für die Projektbeteiligten, sie reduziert sich allerdings darauf, der einen oder anderen Position zur Durchsetzung zu verhelfen oder einen Kompromiß zu finden. Im Falle projekt-interner, antagonistischer Konflikte ist prinzipiell zu bezweifeln, ob ein Projekterfolg überhaupt möglich ist. Floyd u.a. plädieren dafür, zum Beginn von beteiligungsorientierten Software-Projekten einen Projektvertrag zu vereinbaren, der festzuschreiben hätte, daß fundamentale Interessen der Beteiligten nicht verletzt werden. Kommt ein solcher

Projektvertrag nicht zustande, empfehlen sie, das Projekt abzubrechen. [Floyd 1994] In kommerziellen Projekten erscheint ein solches Vorgehen schwer umsetzbar, weil eine solche Entscheidung sich in der Regel dem Zugriff der Projektbeteiligten entzieht. Dennoch ist der darin enthaltene Grundgedanke wichtig, daß nur ein Konsens aller Projektbeteiligten die Grundlage für ein erfolgreiches Projekt sein kann. Dort aber, wo Interessen von Beteiligten antagonistischen Charakter aufweisen, ist es zumindest notwendig, diese offen zu thematisieren.

3.3 Software-Entwicklung im Spannungsfeld konsensual-lösbarer Konflikte

Konflikte, die auf unterschiedlichen Sichtweisen oder auch Mißverständnissen der Beteiligten beruhen und im Verlauf eines gemeinsamen Prozesses durch einen Konsens gelöst werden können, sind konsensual-lösbare Konflikte. In diesen Konflikten weisen die Beteiligten gleiche oder zumindest ähnliche Interessen auf. Ein Konsens ist daher möglich, ohne daß einer der beiden Beteiligten grundlegende Interessenpositionen aufgeben muß. Die soziale Dynamik von Software-Projekten basiert im wesentlichen auf dem Austragen von projekt-internen und projekt-externen Konflikten, die durch einen Konsens lösbar sind. Die zum Konflikt gebrachten Auffassungen reflektieren dann jeweils unterschiedliche Perspektiven der Betrachtung eines Gegenstands. Das Zusammenbringen dieser unterschiedlichen Standpunkte zu einem gemeinsam getragenen dritten Standpunkt, einem Konsens, ist die eigentliche »Triebfeder« der Kooperationsbeziehungen von Software-Projekten.

Weil Software-Projekte auf die Neugestaltung von Arbeitssystemen zielen, berühren sie in besonderer Weise die Interessen der Beschäftigten. Die bei der Software-Entwicklung auftauchenden Probleme lassen sich dabei meist nicht anhand technischer Kriterien lösen und erfordern vielmehr den sozialen Austausch der Beteiligten. Das Fehlen eines »one-best-way«, der sich objektiv bereits am Anfang eines Projekts bestimmen läßt, macht die gemeinsame Suche nach Lösungen unter bewußter Verschränkung der jeweiligen Standpunkte und Perspektiven der einzelnen notwendig. [Floyd 1994] Hierin liegt die Notwendigkeit zum Austragen von Konflikten in modernen Software-Projekten begründet.

Gleichzeitig ist das Austragen von Konflikten die Basis für gemeinsame Lernprozesse zwischen Entwicklern und Anwendern. Huber [1987], Johnson und Johnson [1979] sowie Miller [1986] verweisen darauf, daß interindividuelle Kon-

flikte und Kontroversen[173] als wesentliche Bedingung für die Entfaltung von grundlegenden individuellen und kollektiven Lernprozessen anzusehen sind.[174] Bezogen auf die Lernprozesse im Software-Projekt verweist Reisin darauf, daß die Relativität der Eigenperspektive in kooperativen Zusammenhängen eher erfahrbar wird, weil aufgrund der erforderlichen intersubjektiven Rückkopplung die Notwendigkeit entsteht, die eigene Sichtweise darzustellen und im Verhältnis zu anderen Perspektiven zu identifizieren.[175] In diesem Sinne sind Konflikte die Grundlage kooperativer Lernprozesse.[176]

Der Umgang mit Konflikten im Software-Projekt ist in hohem Maße durch die »Projekt-Kultur« bestimmt. Dort, wo dem Austragen von Konflikten nicht die notwendige Aufmerksamkeit entgegengebracht wird oder Konflikte sogar als kontraproduktiv für den Erfolg aufgefaßt werden, entsteht eine **»Projekt-Kultur«**, die das Finden von gemeinsamen Lösungen zur Gestaltung erschwert. Erfolgreiche Software-Entwicklungen benötigen demgegenüber eine Projekt-Kultur, die den Konflikt zwischen den Beteiligten als notwendig für den gemeinsamen Projekterfolg bejaht und den konstruktiven Umgang mit ihnen aktiv fördert.

4 Kommunikation in der Software-Entwicklung

Kommunikationsprozesse sind für das Gelingen der Software-Entwicklung von grundlegender Bedeutung. Sie dienen im Projekt der Vermittlung der Projektaktivitäten mit der Projektumwelt, der Koordinierung einzelner Aktivitäten der Be-

---

[173] Unter einer Kontroverse wird nach Johnson/Johnson [1979] eine Situation verstanden, in der Schlußfolgerungen, Theorien oder Meinungen zweier Personen unvereinbar sind, sie aber um Übereinstimmung bemüht sind.

[174] "Ausschlaggebend für einen kooperativen Lernprozeß ist die Bereitschaft und Fähigkeit der Kontrahenten, kommunikativ eine Übereinstimmung zu finden, wozu ein gemeinsames Interesse an einer kollektiven Lösung unerläßlich ist." [Reisin 1992, S. 134]

[175] "Für den Einzelnen kann dabei die Gültigkeit seines bereits erworbenen Wissens relativiert oder erschüttert werden. Gleichzeitig kann er sich in den wechselseitigen kommunikativen Handlungen Wissen aneignen, das seinen aktuellen Wissenshorizont überschreitet." [Reisin 1992, S. 136]

[176] "In Kommunikationsprozessen und gemeinsamen Lernprozessen versuchen die Beteiligten offene Probleme und strittige Fragen zu identifizieren und eine gemeinsam akzeptierte Menge von Lösungen zu entwickeln. Indem die Benutzer und Entwickler ihre Widersprüche artikulieren und austragen, bauen sie kommunikativ einen neuen gemeinsamen Geltungsbereich auf, eine gemeinsame Theorie über den zukünftigen Realitätsbereich, in der unterschiedliche Interessen, Anliegen und individuelle Perspektiven ins kollektiv Geltende transzendiert werden." [Reisin 1992, S. 138]

teiligten und der Schaffung und Verstetigung der Identität der Gruppe.[177] Dabei liegt die zentrale Bedeutung der Sprache für die Software-Entwicklung generell darin, daß jegliches soziales Handeln erst über die Sprache begrifflich gefaßt und damit der menschlichen Erkenntnis zugänglich wird. [Paetau 1990, S. 166] Für die Kooperationsbeziehungen im Projekt hat die Kommunikation eine »Querschnittsfunktion«, weil alle sozialen Beziehungen in erster Linie über Sprache vermittelt werden. Die gemeinsamen Lernprozesse zwischen Entwicklern und Anwendern basieren auf der Kommunikationsbeziehung zwischen den beiden Gruppen. Die Regulierung der Konflikte im Projekt ist nur auf der Basis sprachvermittelten Handelns denkbar.

4.1 Kommunikation und Verständigung

Allgemein wird in der technisch orientierten Diskussion »Kommunikation« als Austausch von Informationen zwischen zwei unterschiedlichen Elementen, zwischen denen eine Beziehung existiert, charakterisiert. Die physikalische Basis der Kommunikation ist die Übertragung eines Signals von einer Quelle über einen Sender oder Kanal zu einem Empfänger [vgl. Eco 1987, S. 28] Ausgehend von dieser physikalischen Basis der Kommunikation läßt sich schrittweise das Wesen sozialer Kommunikation erläutern. Die wesentliche Aspekte der Kommunikation sind:

a) Jede Übertragung bedarf einer physikalischen Gestalt, einem **Signal**. Signale können Lichtwellen, magnetische Felder oder elektronische Spannungen sein. Die Art des jeweiligen Signals hängt von der Sender-Empfänger-Beziehung ab.

b) Aus einem Signal wird ein **Zeichen** bzw. ein **Symbol**, wenn es mit einem Signifikationsprozeß korreliert. [Eco 1987] Auf diese Weise wird es zum Träger einer Bedeutung (Signifikanten). Als Träger einer Bedeutung ist es im Kommunikationsprozeß eine **Nachricht**, d.h. "eine abstrakte Form eines Tatbestandes, die zur Speicherung oder Übertragung an einen Dritten dient". [Paetau, S. 79]

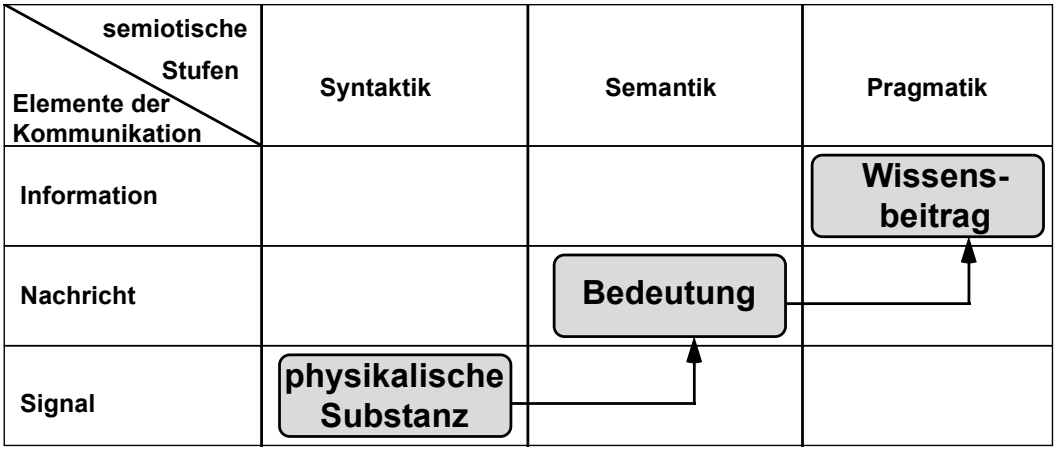
c) Damit aus Nachrichten **»Informationen«** werden, bedürfen sie der Entschlüsselung. Die übertragene Nachricht muß für einen Empfänger informativ sein. Informativ ist sie dann, wenn sie nicht nur einen »Bedeutungsinhalt« hat, sondern vom Empfänger auch in seinen Handlungskontext eingebracht werden

---

[177] Die besondere Bedeutung der Kommunikation in Kooperationsbeziehungen hebt beispielsweise Falck [1992] hervor: "Durch Kommunikation werden somit sachbezogene und soziale Beziehungen gestiftet, reguliert und reproduziert, die es ermöglichen, uns in der Gemeinschaft zu orientieren und variable Verhaltensweisen in unserer Tätigkeit zu entwickeln." [S. 160]

kann, also einen »Sinnbezug« hat.[178] In einer Stufenfolge läßt sich Information wie folgt darstellen:

Stufen des Informationsbegriffs



Quelle: Paetau 1990, S. 81, in Anlehnung an Klaus 1972 und Brönimann 1970

d) Diese informationsübertragenden Beziehungen sind allgemein dann als »**Kommunikation«** zu bezeichnen, wenn zum Aspekt der Informationsverarbeitung der Rückkopplungseffekt [Friedrich 1980, S. 144] oder Reflexivitätsaspekt [Merten 1977, S. 86f. und S. 169f.] kommt. Dann ist nicht mehr eine einzelne Kommunikationsinstanz zu betrachten, sondern zwei sich aufeinander beziehende Instanzen. [Paetau 1990, S. 80]

4.2 Perspektivenverschränkung als Grundlage der Verständigung

Diesem Rückkopplungsaspekt kommt in zwischenmenschlichen Kommunikationsbeziehungen eine besondere Bedeutung zu. Menschen setzen sich mit ihrer Umwelt nicht in Form einer bloßen Reaktion auf vorgegebene Reize auseinander. Menschliches Handeln ist vielmehr das Ergebnis einer aktiven Auseinan-

[178] Das Verhältnis von »Bedeutungsinhalt« und »Sinnbezug« erläutert Paetau an folgendem Beispiel: "Wenn ich beispielsweise bei einer Autofahrt an einer Ansammlung von Häusern vorbeikomme, das Ortsschild jedoch übersehen habe, besitzt der Zeigerstand auf dem Tachometer »80« einen nur geringen Informationswert. Nur wenn ich weiß, daß ich mich in einer geschlossenen Ortschaft befinde, sagt er mir: »Achtung, du bist zu schnell«." [Paetau 1990, S. 80]

dersetzung mit den sozialen und natürlichen Bedingungen des Handelns, d.h. sie reflektieren ihre Handlungssituation, setzen sich bewußt ins Verhältnis zu ihrer Umwelt und handeln auf Grundlage von entwickelten Plänen. Dieses "Sich-bewußt-ins-Verhältnis-setzen" basiert auf der Fähigkeit des gedanklichen "Neben-sich-selbst-Treten" [Mead].

Im zwischenmenschlichen Kommunikationshandeln führt dieses, für das Zustandekommen von Verständigung wichtige »Neben-sich-selbst-Treten« zu einer komplexen sozialen Situation. Denn in der Kommunikation treten Menschen nicht nur in Beziehung zum jeweils anderen und zu sich selbst. Sie beziehen sich in ihren Handlungen vielmehr auf das Bild, welches sie vom anderen haben. Wenn also zwei Kommunikationspartner zueinander in Beziehung treten, ergibt sich formal betrachtet folgende Beziehung:

Die Partner x und y haben jeweils ein bestimmtes Selbstbild x' und y' als auch ein bestimmtes Bild vom Anderen (x'' und y''). Im gemeinsamen Kommunikationsprozeß richtet sich

x als x' an y'',

während

y als y' gegenüber x'' reagiert.

Die Kommunikationsbeziehung zwischen x und y verläuft demnach so, daß die mitgeteilten Nachrichten immer in Anbetracht der konkreten Beziehung zwischen den jeweiligen Gesprächspartnern spezifiziert werden. Unter diesen Bedingungen kann Verständigung in der Kommunikation nur gelingen, wenn es den Gesprächspartnern im Verlaufe des Gesprächs gelingt, ihre jeweiligen Sichtweisen zu verschränken. Diese gegenseitige Perspektivenverschränkung basiert nach Mead auf einer gegenseitigen "Rollenübernahme" ("role-taking"), also auf dem Prozeß des Sich-hinein-Versetzens in die Rolle des Anderen. Dabei nimmt der Handelnde die "Haltung der anderen gegenüber sich selbst und jeder gesellschaftlichen Situation ein, in der er und andere Menschen gestellt oder impliziert sein können". [Mead 1968, S. 281]

Betrachtet man diese gegenseitige Perspektivenverschränkung in einer Gruppe, so reicht es für den einzelnen nicht mehr aus, ein Verhältnis zwischen sich, seiner Rolle und einem anderen Gesprächspartner herzustellen. Vielmehr müssen sich alle Gruppenmitglieder am Verhalten aller orientieren, damit eine Verständigung zustande kommt. Dabei haben die für alle Gruppenmitglieder gültigen Ziele, Werte und Normen eine besondere Bedeutung. In diesem Normensystem spiegelt sich eine verallgemeinerte Verhaltenserwartung der Gruppenmitglieder, nach Mead der "verallgemeinerte Andere", wider. [Mead 1968, S. 196]

Bei zwischenmenschlicher Kommunikation vollzieht sich die Verständigung über die wechselseitige Perspektivenverschränkung. Diese erfordert ein Hineinversetzen in die Rolle des anderen. Dies geschieht zunächst auf der Basis des Bildes, das von dem Partner besteht und während der Kommunikation verändert oder sogar aufgebaut werden kann. Daraus resultiert für das Individuum die Notwendigkeit, ständig zwischen der eigenen Rolle und der der anderen zu wechseln sowie sich selbst als Subjekt zu verhalten und sich gleichzeitig als Objekt in der entsprechenden Beziehung zu betrachten.

Entwickler und Anwender trennen häufig sehr unterschiedliche »Weltbilder« und damit verbundene Wahrnehmungen der betrieblichen Realität. Das erschwert die Herstellung einer erfolgreichen Kommunikationsbeziehung im Software-Projekt. Entwickler stellen häufig technikimmanente Aspekte der Gestaltungsaufgabe in den Vordergrund. Die von ihnen angestrebte Qualität des Produkts bezieht sich vorrangig auf technische Qualitätskriterien. Demgegenüber steht für Anwender vorrangig der Aufgaben- und Anwendungsbezug während des Projekts im Vordergrund. Ihre Ziele waren daher eher »Erleichterung bei der Aufgabenerledigung« oder »Belastungsreduzierung«. [vgl. Heilmann 1981] Erschwert wird die Kommunikationsbeziehung beider Seiten weiterhin durch Vorurteile und falsche Vorstellungen über die jeweils andere Gruppe. Das behindert die Verständigung von Anwendern und Entwicklern im Software-Projekt.

Das Herstellen einer gemeinsamen Verständigung über die im Projekt anstehenden Fragen erfordert die Entwicklung einer gemeinsamen **Projekt-Sprache**, die Entwickler und Anwender gleichermaßen verstehen. Die Basis für die Entwicklung einer gemeinsamen Sprache ist die gemeinsame Erschließung der konkreten Arbeitswelt der Anwender in deren Alltagssprache. Entwickler müssen demnach lernen, ihre Kompetenzen in die Sprache der Anwender zu übertragen. Die Umfeldbedingungen des Projekts und die **Projekt-Kultur** prägen das Zustandekommen von Verständigung zwischen Anwendern und Entwicklern nachhaltig.

5 Kooperationskompetenz in der Software-Entwicklung

Moderne Software-Entwicklung ist als kooperative Gestaltung zu verstehen. Diese kooperative Gestaltungsaufgabe beinhaltet gemeinsame Lernprozesse zwischen Entwicklern und Anwendern, das gemeinsame Lösen von Konflikten und übergreifend die Herstellung einer erfolgreichen Kommunikationsbeziehung. Gestaltung ist vor allem als Theoriebildung zu verstehen. Nicht das technische Erzeugen von Software-Produkten bestimmt demnach den Entwicklungsprozeß in erster Linie, sondern die kooperative Erarbeitung einer gemeinsamen Theorie über die Gestaltungsaufgabe. Die Realisierung und die Implementierung von

Software-Systemen ist dem logisch nachgeordnet. Kooperative Gestaltung läßt sich insgesamt unter den Aspekten

- »Herstellung einer gemeinsamen Lernbeziehung«,
- »Lösen von Konflikten«,
- »Herstellung einer erfolgreichen Kommunikationsbeziehung« und
- »Verständigung über eine gemeinsame Theorie des künftigen Arbeitsprozesses« betrachten.

Das methodische Vorgehen, das im Projekt jeweils gewählt wird, hat diesem besonderen Charakter der Gestaltungsaufgabe gerecht zu werden. Nicht die Orientierung an der Erstellung des technischen Produkts, wie es vom traditionellen Software Engineering vorgeschlagen wird, darf im Vordergrund stehen, sondern die Entfaltung der Dynamik des Kooperationsprozesses im Software-Projekt. Der Einsatz von Methoden und Techniken zur Erstellung des Software-Produkts ist daher den Kooperationsmethoden nachgeordnet.

Für die Auswahl des geeigneten methodischen Vorgehens im Software-Projekt entspringen aus der Tatsache, daß hier unterschiedliche Gruppen kooperieren, zwei übergeordnete Anforderungen.

Die gewählten Methoden müssen eine **gemeinsame Verständigung** von Anwendern und Entwicklern prinzipiell ermöglichen und fördern. Herkömmliche Beschreibungsmethoden wie Entity-Relationship-Modelle oder SADT-Aktigramme sind zwar für die Verständigung der Entwickler untereinander oft wichtig. Sie sind aber für die Zusammenarbeit mit den Anwendern nicht geeignet, weil diese in der Regel über keine Erfahrungen im Umgang mit diesen Methoden verfügen. Die Zusammenarbeit zwischen Entwicklern und Anwendern wird durch die Verwendung dieser Methode nicht gefördert, weil beide Gruppen sie nicht in gleichem Maße handhaben können.

Weiterhin gilt, daß das gewählte Vorgehen für Entwickler und Anwender gleichermaßen verständlich sein muß. Dies ist nur mit Methoden möglich, die weitgehend auf der konkreten Erfahrungswelt der Anwender aufbauen und auf deren Alltagssprache basieren.

Die gewählten Methoden müssen weiterhin die **konstruktive Lösung von Konflikten** im Projekt ermöglichen. Anwender und Entwickler bringen jeweils sehr unterschiedliches Vorwissen und jeweils besondere Erfahrungen in die Zusammenarbeit ein. Diese korrespondieren oft mit unterschiedlichen Auffassungen über die eigene Rolle im Entwicklungsprozeß und entsprechenden Rollenerwartungen gegenüber der jeweils anderen Gruppe. Hierin wurzeln Konflikte im

Projekt. Aufgrund der betriebspolitischen Bedeutung von Software-Projekten wirken sie nachhaltig auf das Geflecht von Interessen und Macht, das in Unternehmen besteht. Interessenauseinandersetzungen im Projekt lassen sich daher nicht umgehen.

Gleichzeitig ist das Austragen dieser in unterschiedlichen Interessen wurzelnden Konflikte eine wichtige Grundlage für gemeinsame Lernprozesse im Projekt. Nur im gemeinsamen Austragen bestehender Auseinandersetzungen, dem Aufeinanderzubewegen unterschiedlicher Standpunkte wurzeln gemeinsame Lernprozesse von Entwicklern und Anwendern. Das methodische Vorgehen im Entwicklungsprozeß muß der Formulierung von Interessenpositionen und dem Austragen der Konflikte entsprechenden Raum einräumen. Das erfordert Vorgehensweisen, die es Anwendern und Entwicklern ermöglichen, aus kontroversen Standpunkten gemeinsame Entscheidungen zu machen. Die Methoden müssen demnach einen **Beitrag zur Entscheidungsfindung** beinhalten.

Weitere Anforderungen an die Wahl der Methoden entstehen aus der Charakteristik der jeweils zu bearbeitenden Problemstellung.

Die bei der Software-Entwicklung auftauchenden Probleme lassen sich meist nur im sozialen Austausch der Beteiligten lösen. Ihrem Charakter nach sind die Mehrzahl der zu lösenden Probleme im Software-Projekt dialektischer Natur; d.h. solche, bei denen zu Beginn des Problemlösungsprozesses weder die Ist- und Sollkriterien noch die Operatoren, die zu einer erfolgreichen Problemlösung führen, eindeutig bestimmt werden können.[179] Bei diesem Problemtyp lassen sich Lösungen nicht allein durch den Einsatz analytischer Verfahren finden. Die Lösungssuche erfordert darüber hinaus vielmehr den Einsatz von Methoden, die auf heuristischen Prinzipien[180] basieren und insbesondere die Entscheidungsfindung in der Gruppe unterstützen.

Methoden eignen sich entweder eher zur Lösung von analytischen Problemen oder sie leisten einen Beitrag zum Finden von Lösungen bei dialektischen bzw. offenen Problemstellungen. Je nach zu bearbeitendem Problemtyp sind im Software-Projekt daher solche Methoden zu wählen, die entweder die analytische Erschließung eines Problems betonen oder solche, die die heuristische Erschließung ermöglichen.

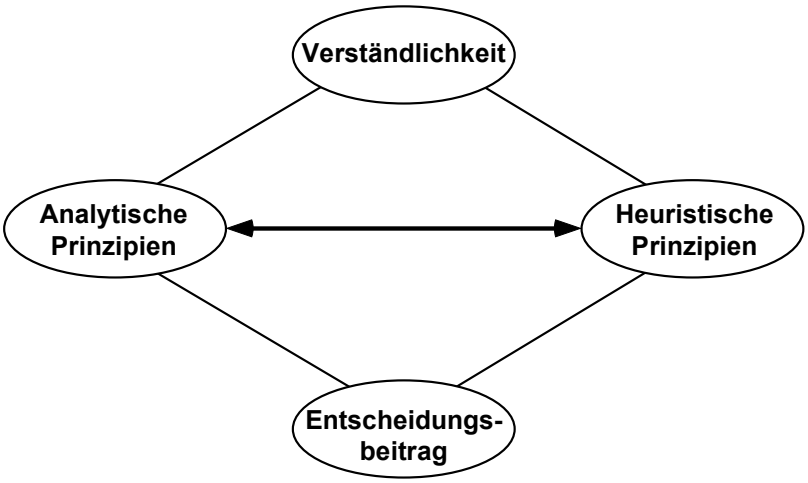
---

[179] Bezogen auf verschiedene Problemtypen entwickeln Sell und Fuchs-Frohnhofen eine Systematik von Problemlösungsmethoden und entsprechende Qualifizierungsvorschläge. [Sell/Fuchs-Frohnhofen 1993, S. 46ff.]

[180] Heuristische Prinzipien sind »Finderegeln«, die die Wahrscheinlichkeit des Findens einer Lösung erhöhen, ohne daß sie eine Erfolgsgarantie beinhalten.

Die Wahl der Methoden steht demnach generell in einem Spannungsfeld unterschiedlicher Anforderungen. Die übergeordnete Bezugsebene resultiert aus dem kooperativen Charakter von Software-Projekten. Die gewählten Methoden müssen für alle Projektbeteiligten gleichermaßen handhabbar und verständlich sein. Und sie müssen die Lösung von Konflikten im Projekt ermöglichen. Die Wahl des methodischen Vorgehens muß darüber hinaus durch den zu lösenden Problemtyp bestimmt sein. Je nach zu lösender Aufgabenstellung sind solche Methoden zu wählen, die eher auf die analytische oder die heuristische Erschließung von Lösungen hinauslaufen.

Methodisches Vorgehen im Spannungsfeld



Die Auswahl und der Einsatz von Methoden ist je nach der Handlungssituation, in der sich das Projekt befindet, differenziert zu betrachten. In Software-Projekten lassen sich verschiedene **typische Handlungssituationen** identifizieren, die jeweils unterschiedliche Anforderungen an die Zusammenarbeit stellen. Diese Handlungssituationen unterscheiden sich hinsichtlich

- der Zielstellung,
- der zu erstellenden Produkte und insbesondere
- der Erfordernisse an den Kooperationsprozeß

von anderen Handlungssituationen. Unabhängig davon, nach welchem Projektmodell ein Projekt bewältigt wird, lassen sich insgesamt vier kritische Handlungssituationen unterscheiden. Diese sind:

1. Die Erarbeitung eines Verständnisses des Arbeitsprozesses,
2. die Auseinandersetzung mit Gestaltungsoptionen,
3. die Entwicklung einer Theorie des zukünftigen Arbeitsprozesses und
4. die Implementierung des entwickelten Software-Produkts.[181]

Erst das Verständnis der Besonderheiten der jeweiligen kritischen Handlungssituation ermöglicht den Entwicklern einen handlungskompetenten Umgang mit den Methoden.

5.1 Kritische Handlungssituation 1: Erarbeitung eines Verständnisses des Arbeitsprozesses

Die Grundlage für die Neugestaltung eines Arbeitsprozesses ist die Erarbeitung eines gemeinsamen Verständnisses des bestehenden Arbeitsprozesses. Hierbei werden die zur Bewältigung der jeweiligen Projektaufgabe wesentlichen Aspekte des bestehenden Arbeitsprozesses in allgemeiner Form erschlossen. Dieser Prozeß hat für die Erfüllung der Projektaufgabe mehrere Funktionen:

- Er ist die entscheidende Basis für Anwender und Entwickler, um aufbauend darauf die Erfordernisse zur Neugestaltung in ausreichender Konkretion beschreiben zu können.
- Er ist für Anwender und Entwickler gleichermaßen die Basis zum gestaltungsorientierten Verständnis der konkreten Arbeitssituation. Die Anwender erarbeiten sich dadurch eine »Draufsicht« auf ihre Arbeitsrealität und erreichen dadurch die Grundlage, um Neugestaltungserfordernisse bestimmen zu können. Die Entwickler gewinnen in diesem Prozeß ein allgemeines Verständnis der konkreten Arbeitssituation als Grundlage für die Entwicklung der Software.
- Beide Seiten entwickeln darüber hinaus eine entsprechende Verständigungsebene in Form einer gemeinsamen Projekt-Sprache.

---

[181] Diese vier Handlungssituationen bilden eine Einheit. Auch die vierte Handlungssituation, die Implementierung, ist als Teil des Entwicklungsprozesses anzusehen und steht in der Verantwortung des Software-Projekts. Für den kooperativen Gestaltungsprozeß sind die ersten drei Handlungssituationen von besonderer Bedeutung. Die methodischen Probleme des Kooperationsprozesses dieser Situationen werden im folgenden näher behandelt.

Je nach Schwierigkeit der Projektaufgabe kann dieser Prozeß in wenigen Gesprächen bewältigt werden und in einige wenige Skizzen oder Aufzeichnungen münden. Er kann aber auch mehrere Wochen oder Monate in Anspruch nehmen und den Einsatz komplexer Methodenapparate zur Erschließung der Realität und zur Dokumentation und Bewertung der Ergebnisse erfordern. In der Projektrealität kann die Erarbeitung eines Verständnisses des Arbeitsprozesses zeitlich eng mit den übrigen kritischen Handlungssituationen - insbesondere der Entwicklung einer Theorie des zukünftigen Arbeitsprozesses und der Auseinandersetzung mit Gestaltungsoptionen - verbunden sein und daher zeitlich kaum von ihnen zu unterscheiden sein. Er kann aber auch als eigenständige Phase »Ist-Analyse« von den übrigen Handlungssituationen zeitlich getrennt werden. Unabhängig von diesen Besonderheiten der Projektabwicklung stellt die Erarbeitung einer abstrakten Beschreibung des Arbeitsprozesses eine spezifische Handlungssituation dar, die in jedem Projekt bewältigt werden muß.

Zentrale Anforderungen an den Kooperationsprozeß

Traditionelle Methoden des Software Engineering behandeln diese Handlungssituation nur am Rande. Hinsichtlich der Erschließung der konkreten Arbeitsrealität wird meist vereinfachend darauf verwiesen, daß über Befragungen oder das Studium von Dokumenten die notwendigen Informationen zum bestehenden Arbeitssystem erreicht werden können. Anwender sind in diesem Projektverständnis nicht aktiv beteiligt, sondern passive Objekte der Beschaffung der erforderlichen Informationen. Für die Dokumentation und Bewertung der Ergebnisse wird auf Beschreibungsmethoden wie Entity-Relationship-Modelle oder SADT-Aktigramme verwiesen. Daß diese Methoden den meisten Anwendern nicht geläufig sind, wird zwar in verschiedenen Modellprojekten zur beteiligungsorientierten Software-Entwicklung als zentrales Problem thematisiert, hat aber für das traditionelle Software Engineering wenig Bedeutung. Denn hier erfolgt die Auswahl der Methoden nicht nach der Maßgabe ihrer Tauglichkeit für die Kooperation zwischen Anwendern und Entwicklern, sondern lediglich hinsichtlich ihrer Brauchbarkeit zur Erstellung des technischen Produkts »Software«. Für das klassische Software Engineering gelten u.a. folgende Prämissen bei der Auswahl der Methoden:

- Die erforderlichen Informationen liegen analytisch erschließbar vor. Sie sind entweder über Dokumente ermittelbar, oder sie sind den Anwendern vollständig bewußt, so daß sie den Entwicklern in ausreichender Genauigkeit auf einfaches Befragen hin mitgeteilt werden können.

- Die Anwender sind dabei ohne weiteres bereit, diese Informationen an die Entwickler preiszugeben, ohne ihrerseits die weitere Verwendung dieser Informationen mitbestimmen zu wollen.
- Die Entwickler sind ohne intensive Zusammenarbeit mit den Anwendern in der Lage, deren konkrete Arbeitssituation ausreichend genau zu verstehen.
- Einmal erhobene Informationen müssen entweder nicht mehr mit den Anwendern hinsichtlich ihrer Vollständigkeit und Richtigkeit bewertet werden, oder diese Aufgabe läßt sich anhand der klassischen Beschreibungsmethoden durchführen.

Diese Prämissen sind in modernen Projekten kaum noch gegeben. Ein hinreichendes Verständnis des Arbeitsprozesses läßt sich hier nur in direkter Kooperation von Anwendern und Entwicklern erreichen. In dieser kritischen Handlungssituation sind folgende zentrale Anforderungen gestellt:

a) In einem gemeinsamen Prozeß erarbeiten sich Anwender und Entwickler das Arbeitssystem der Anwender als gemeinsamen Realitätsbereich. Diese Erschließung des Arbeitsprozesses, also der Arbeitstätigkeiten, -mittel und -gegenstände sowie der Kooperationsbeziehungen erfolgt als gemeinsame Rekonstruktion, aufbauend auf der Alltagserfahrung der Anwender. [Reisin 1992] Als Grundlage aller weiteren Gestaltungsentscheidungen müssen die hier gewonnenen Beschreibungen den zu gestaltenden Arbeitsprozeß mit ausreichender Genauigkeit wiedergeben. Dies erfordert von Anwendern und Entwicklern ein Vorgehen im Projekt, das auf eine exakte Erschließung des Realitätsbereichs gerichtet ist.

Genauigkeit ist die erste Anforderung an das Vorgehen in dieser kritischen Handlungssituation.

b) Die Arbeitsrealität der Anwender läßt sich nicht durch das Studium von Dokumenten und Unterlagen erschließen. Vorhandene Beschreibungen wie Stellenbeschreibungen oder Organigramme, aber auch das in Interviews abfragbare Wissen von Vorgesetzten, vermitteln bestenfalls eine sehr grobe Vorstellung von dieser Realität. Erschließen läßt sich das Arbeitssystem nur über die Erfahrungen der Anwender. Diese Erfahrungen liegen den Anwendern aber meistens nicht in einer Form vor, die sie den Entwicklern einfach mitteilen können. Sie sind nur zu einem kleinen Teil im Sinne von regelhaftem Wissen formulierbar. Der Großteil des im konkreten Arbeitsprozeß erworbenen Wissens ist als »Erfahrungswissen« vorhanden. Über dieses Wissen wird nicht in Form von Regeln oder klar umrissenen Begriffen verfügt. Um die Anwender zu unterstützen, sich dieses Erfahrungswissen zu erschließen, ist ein Vorgehen notwendig, das ihnen hilft, sich über Assoziationen ihren Erfahrungen zu nähern.

Die zweite Anforderung an das Vorgehen in dieser kritischen Handlungssituation heißt: Seitens der Entwickler muß Wert darauf gelegt werden, den Anwendern - gestützt auf heuristische Verfahren - Gelegenheit zum assoziativen Erschließen ihres Erfahrungswissens zu geben.

Methodisches Vorgehen

Die Auswahl der Methoden und Techniken zur Erarbeitung einer abstrakten Beschreibung des Arbeitsprozesses ist generell an der übergeordneten Zielstellung der Unterstützung der Kooperation im Projekt zu orientieren. Das methodische Vorgehen in dieser kritischen Handlungssituation hat darüber hinaus ein Spannungsfeld folgender Anforderungen zu beachten:

a) Das gewonnene Verständnis muß den Arbeitsprozeß mit ausreichender **Genauigkeit und Vollständigkeit** erfassen. Das erfordert Methoden, die eine exakte Erschließung des konkreten Arbeitsprozesses erlauben.

b) Andererseits müssen die Anwender durch ein assoziationsbetontes Vorgehen dabei unterstützt werden, sich ihr Erfahrungswissen zu erschließen, um es mitteilen zu können. Das erfordert Methoden, die Raum für Assoziationen lassen und geradezu auf **»Ungenauigkeit«** aufbauen.

Diese beiden Anforderungen an das methodische Vorgehen stehen in einem widersprüchlichen Verhältnis zueinander. Zur Verfügung stehende Methoden werden entweder mehr dem einen oder dem anderen Anspruch gerecht. Das bedeutet, daß in der kritischen Handlungssituation zur Erarbeitung einer abstrakten Beschreibung des Arbeitsprozesses sowohl Methoden, die auf analytischen Prinzipien basieren als auch solche, die auf heuristischen Prinzipien aufbauen, zum Einsatz kommen müssen. Die Schwierigkeit besteht darin, einen der jeweiligen Handlungssituation angemessenen Methoden-Mix zusammenzustellen.

Für die abstrahierende Erschließung und Beschreibung des Arbeitsprozesses können folgende Methodenarten eingesetzt werden:

- Dokumentenanalyse,
- Beobachtungsmethoden,
- Befragungsmethoden,
- Besprechungs- und Diskussionsmethoden.

5.1.1 Dokumentenanalyse

Die Analyse von Dokumenten[182] ist eine wichtige Informationsquelle zur Erschließung eines konkreten Arbeitsprozesses. Diese Methode stützt sich in dieser Handlungssituation auf die Analyse von

- Unterlagen über das Arbeitssystem, wie Organigramme, Stellenbeschreibungen oder Arbeitsanweisungen,
- Dokumenten des Arbeitsprozesses, wie Rechnungsformulare, Briefvorlagen oder Kundendateien,
- Fachbüchern und Hintergrundinformationen über fachliche Fragen des konkreten Arbeitsprozesses.

Die Dokumentenanalyse ermöglicht den Entwicklern eine erste Orientierung über den konkreten Arbeitsprozeß, um aufbauend darauf die Gespräche mit Anwendern vorbereiten zu können. Der alleinige Einsatz der Dokumentenanalyse reicht heute meist nicht mehr aus, um die notwendigen Informationen über ein Arbeitssystem zu erhalten. Daher wird diese Methode häufig in Kombination mit anderen Vorgehensweisen eingesetzt.

Zusammenfassend läßt sich die generelle Eignung der Methode wie folgt einschätzen:

- Da sich die Methode auf Dokumente stützt, die den Anwendern meist bekannt sind, erfüllt sie das Verständlichkeitskriterium in diesem Punkt. Für die Schaffung einer gemeinsamen Verständigungsbasis bietet dieses Vorgehen lediglich eine Grundlage.
- Der Einsatz der Dokumentenanalyse eignet sich lediglich zur Informationsbeschaffung. Die Methode beinhaltet keinen Entscheidungsbeitrag im Kooperationsprozeß von Entwicklern und Anwendern.
- Im isolierten Einsatz bietet die Dokumentenanalyse keinen Raum für die Unterstützung von Assoziationsprozessen der Anwender. Der Kontakt mit den Anwendern beschränkt sich hier meist auf das Beschaffen der erforderlichen Unterlagen. Wird die Analyse von Dokumenten aber mit anderen Methoden wie Befragungen oder Gruppendiskussionen kombiniert, so bietet sie eine gute Grundlage für die assoziative Erschließung der Erfahrungen der Anwender.

---

[182] Die Dokumentenanalyse ist eine klassische Methode der empirischen Forschung. Eine umfassende Beschreibung dieser Methode findet sich in den einschlägigen Handbüchern.

5.1.2 Beobachtungen

Beobachtungsmethoden[183] kommen bei der Erarbeitung von abstrakten Beschreibungen des Arbeitsprozesses insbesondere zur Erschließung von konkreten Arbeitsabläufen und -aufgaben zum Einsatz. Hierbei beobachten die Entwickler die Anwender bei der Arbeit, um sich die konkrete Arbeitssituation insgesamt oder in Teilen zu erschließen. Nimmt der beobachtende Entwickler an dem Arbeitsprozeß teil, spricht man von einer teilnehmenden Beobachtung. Steht der Entwickler außerhalb des Beobachtungsfeldes, so handelt es sich um eine nichtteilnehmende Beobachtung. Zur Dokumentation der Ergebnisse werden neben vorher ausgearbeiteten Dokumentationsunterlagen häufig Videoaufzeichnungen genutzt.

Beobachtungen sind seitens der Entwickler für folgende Erhebungszwecke einsetzbar:

- Um einen unmittelbaren Eindruck der konkreten Arbeitssituation zu erhalten. So sind z.B. Betriebsbegehungen zu Beginn eines Projekts von sehr großer Bedeutung, um ein Verständnis der Arbeit zu erhalten.
- Um mit anderen Methoden gefundene Informationen zu überprüfen. So kann z.B. die Frage, ob ein auftretendes Problem mit dem Computer auf Bedienfehler oder ein fehlerhaftes Programm zurückzuführen ist, oft nur durch eine Beobachtung geklärt werden.
- Um konkrete Arbeitsabläufe mit großer Genauigkeit studieren zu können. Da für die Anwender viele Arbeitsabläufe routinisiert ablaufen, lassen sich durch Befragen die Tätigkeitsabläufe oft nicht genau und vollständig rekonstruieren. Beobachtungen liefern hier oft bessere Ergebnisse.

Beobachtungen sind im isolierten Einsatz auf die Erschließung solcher Erkenntnisse beschränkt, die sich durch die unmittelbare Anschauung gewinnen lassen. Der beobachtende Entwickler sieht demnach nur, was ein Anwender tut, weiß damit aber noch nicht, was dieser sich dabei denkt und warum er es tut. Die meisten Arbeitshandlungen lassen sich für Außenstehende aber erst dann

---

[183] Die Beobachtung ist eine klassische Methode der empirischen Forschung. Eine umfassende Beschreibung dieser Methode findet sich in allen einschlägigen Handbüchern. Sehr weit ausgearbeitete Leitfäden zur qualifizierten Erschließung von Arbeitssystemen durch Beobachtungen finden sich in der arbeitspsychologischen Forschungsliteratur. [vgl. Hacker u.a. 1989, Oesterreich/Volpert 1991]. Diese Methoden erfordern meist eine Grundqualifikation und sind für Laien nicht erfolgreich einsetzbar. Sie können aber als Grundlage dienen, um eigene Beobachtungsinstrumente zu erstellen.

erschließen, wenn der Handlungskontext - also das »Warum?« und das »Wozu?« der Handlung - bekannt ist. Dieser Tatbestand schränkt die Einsetzbarkeit und die Genauigkeit von Beobachtungsmethoden ein. Beobachtungsmethoden werden daher zur Erschließung der Arbeitssituation oft mit Befragungen oder der Technik »Lautes Denken« verbunden, um dieser Einschränkung entgegenzuwirken. In beiden Fällen werden die Anwender von den Entwicklern begleitend zur Beobachtung nach näheren Auskünften über ihre Arbeit befragt oder aufgefordert, ihre Gedanken während des Arbeitens auszusprechen.

Zusammenfassend läßt sich die generelle Eignung der Methode wie folgt einschätzen:

- Soweit der Einsatz dieser Methode keinen aktiven Austauschprozeß zwischen Entwickler und Anwender beinhaltet, ist die Verständlichkeit nicht erforderlich. Beim kombinierten Einsatz der Beobachtung mit Befragungen o.ä. ist eine Verständigungsbasis vorhanden, weil sich die Erhebung auf den Arbeitsablauf der Anwender bezieht.
- Der Einsatz der Beobachtung eignet sich lediglich zur Informationsbeschaffung. Die Methode beinhaltet aber keinen Entscheidungsbeitrag im Kooperationsprozeß von Entwicklern und Anwendern.
- Im isolierten Einsatz erhält die Beobachtungsmethode in der hier zu bearbeitenden kritischen Handlungssituation keinen Assoziationsbeitrag. Die Anwender leisten keinen aktiven Beitrag zum Zustandekommen des Ergebnisses, sie sind lediglich Objekte der Beobachtung. In Kombination mit Befragungsmethoden oder »Lautem Denken« entsteht für die Anwender ein Assoziationsspielraum, der es ihnen ermöglicht, sich Erfahrungswissen zu erschließen.[184]

5.1.3 Befragungsmethoden

Befragungsmethoden[185] haben bei der Erschließung eines Arbeitsprozesses eine erstrangige Bedeutung. Sie sind ohne größere Vorbereitungen einsetzbar, stoßen bei Fragern und Befragten auf relativ große Akzeptanz und werden daher

---

[184] Hacker verweist beim kombinierten Einsatz von Beobachtungen und Befragungen auf positive Erfahrungen bei der Erschließung von Expertenwissen. [Hacker 1992]

[185] Die Befragung ist als klassische Methode der empirischen Forschung in allen einschlägigen Handbüchern. Als inhaltliche Grundlage zur Erarbeitung von Fragestellungen können arbeitspsychologische Instrumente und die in Kapitel 3 dieses Curriculums entwickelte Systematik dienen. Bei Holz auf der Heide [1993] findet sich eine bewertende Beschreibung des Einsatzes von Befragungsmethoden zur Analyse eines Arbeitssystems.

in Software-Projekten oft verwandt. Durch schriftliche oder mündliche Befragungen einzelner Anwender oder Anwendergruppen erhalten die Entwickler wichtige Informationen über das konkrete Arbeitssystem. Je genauer sie dabei die Antwortmöglichkeiten eingrenzen können, desto geschlossener können die Fragestellungen formuliert sein. Im Extrem lassen sie sich auf die Möglichkeiten »Ja« oder »Nein« beschränken. Sind demgegenüber die möglichen Antworten dem Fragenden im Vorhinein nicht bekannt oder wird bei der Befragung Wert auf die subjektive Einschätzung der Befragten gelegt, sind die Fragen hinsichtlich ihrer Antwortmöglichkeiten entsprechend offen zu formulieren.

Befragungen sind seitens der Entwickler für folgende Zwecke einsetzbar:

- Um sich einen ersten Einblick in das konkrete Arbeitssystem zu verschaffen. Hierfür eignen sich offene, mündliche Befragungen von Personen, die einen Überblick geben können.
- Um wesentliche Aspekte des Arbeitsprozesses oder einzelner Arbeitsaufgaben genauer zu erschließen. Hierfür eignen sich halb-offene, mündliche Interviews.
- Um bestimmte Informationen wie Anzahl der Mitarbeiter, Varianten eines Produkts usw. gezielt zu erfragen. Je nach Stand der Vorkenntnisse und des Fragegegenstands sind die Fragen halb-offen oder geschlossen zu formulieren.
- Um Einschätzungen und Meinungen einzelner Beschäftigter oder ganzer Gruppen einzuholen. Hierfür eignen sich offene oder halb-offene Befragungen. Je nach Anzahl der befragten Personen und den statistischen Erfordernissen sind mündliche oder schriftliche Formen zu wählen.

Zusammenfassend läßt sich die generelle Eignung der Methode wie folgt einschätzen:

- Befragungsmethoden erfüllen bei entsprechender Formulierung der Fragen das Verständlichkeitskriterium.
- Der Einsatz der Befragungen eignet sich vorrangig zur Informationsbeschaffung. Die Methode beinhaltet aber keinen direkten Entscheidungsbeitrag im Kooperationsprozeß von Entwicklern und Anwendern. Durch Erfragen von Meinungen und Einschätzungen können die Entscheidungsprozesse durch den Einsatz dieser Methode allerdings vorbereitet werden.
- Je nach Fragetechnik bieten sie die Möglichkeit, die Anwender nicht nur nach Aspekten zu befragen, die ihnen unmittelbar bewußt sind. Sie können

aber auch eingesetzt werden, um die Anwender bei der assoziativen Erschließung ihrer Erfahrungen zu unterstützen.

5.1.4 Besprechungs- und Diskussionsmethoden

Gruppendiskussionen, Besprechungen und Workshops[186] haben in allen Handlungssituationen des Software-Projekts eine hervorgehobene Bedeutung, weil sie eine gute Grundlage zur Organisation der Zusammenarbeit im Projekt bieten. Sie sind insbesondere geeignet, die soziale Dynamik von gemeinsamen Lernprozessen zu fördern und bieten Raum, um Konflikte auszutragen. Für die kooperative Bewältigung der Projektaufgabe sind sie daher generell unverzichtbar. Bei der Erarbeitung einer abstrakten Beschreibung des Arbeitsprozesses sind Besprechungs- und Diskussionsmethoden darüber hinaus vor allem einsetzbar, um

- neue Informationen über das Arbeitssystem zu erschließen,
- gefundene Informationen zu bewerten und
- Entscheidungen in der Gruppe zu fällen.

Je nach Ziel der Sitzung variiert die Zusammensetzung der Gruppe, deren Rollenverteilung, das Thema und das methodische Vorgehen.

Wenn Gruppendiskussionen zur Erschließung des Arbeitssystems eingesetzt werden oder die Bewertung von bereits gewonnenen Informationen zum Gegenstand haben, wird eine festzulegende Anzahl von Anwendern eingeladen, um eine bestimmte Fragestellung zu diskutieren. Hierbei nehmen die Entwickler die Rolle von Diskussionsleitern, Moderatoren oder Beobachtern ein. Ihre Aufgabe besteht darin, die Diskussion vorzubereiten, anzuregen und zu dokumentieren. Wenn die Methode eingesetzt wird, um Entscheidungen zu treffen, sind die Entwickler über die oben benannten Aufgaben hinaus oft auch direkt als aktive Teilnehmer an den Sitzungen beteiligt.

Je nach zur Debatte stehendem Problem können unterschiedliche Methoden zur Unterstützung der Entscheidungsprozesse zur Hilfe genommen werden. Schlicksupp unterscheidet systematisch-analytische Techniken von solchen Verfahren, die die Kreativität und Intuition fördern. Die erste Gruppe von Tech-

---

[186] Eine gut lesbare allgemeine Einführung zu Besprechungs- und Diskussionsmethoden, sowie darin eingebetteter Kreativitäts-, Konfliktlösungs- und Entscheidungsfindungstechniken findet sich bei Brühweiler [1992] und Schlicksupp [1983]. Für das Erlernen dieser Methoden im Unterricht ist allerdings - wie bei allen hier vorgeschlagenen Methoden - eine Operationalisierung für die konkreten Handlungssituationen notwendig.

niken dient der systematischen Erfassung, Ordnung und Gliederung von Aspekten eines Problems und organisiert die Lösungssuche durch systematische Kombination und Variation. Die zweite Gruppe orientiert auf die Erschließung des kreativen Potentials des Projektteams und fördert die Lösungssuche durch die Unterstützung der intuitiven Fähigkeiten. [vgl. Schlicksupp 1983, S. 35ff.]

Die Besprechungs- und Diskussionsmethoden bieten gute Grundlagen, um die konkreten Erfahrungen der Anwender zu erschließen. Sie sind für die Entwickler darüber hinaus unverzichtbar, um sich in Kommunikationsprozessen mit den Anwendern ein konkretes Verständnis der Arbeitssituation zu erarbeiten. Außerdem sind sie die entscheidende methodische Grundlage für das Herstellen eines gemeinsamen Verständnisses zwischen beiden Gruppen. Da sie auf dem Medium Sprache basieren, stellen Besprechungs- und Diskussionsmethoden hohe Anforderungen an die Dokumentation von Ergebnissen und Zwischenergebnissen. Eine Kombination dieser Methoden mit Visualisierungs- oder Moderationstechniken [vgl. Schlicksupp 1983] ist unbedingt erforderlich.

5.1.5 Zusammenfassung

Das methodische Herangehen in der kritischen Handlungssituation zur Erarbeitung einer abstrakten Beschreibung des Arbeitsprozesses stellt folgende Anforderungen an das methodische Vorgehen:

- Die Erschließung der notwendigen Informationen erfordert sowohl den Einsatz von Methoden, die die Informationsbeschaffung mit großer Genauigkeit ermöglichen, als auch solche, die den Anwendern Raum bieten, um sich ihre Erfahrungen zu erschließen. Keine der genannten Methodenarten erfüllt dieses Kriterium alleine. Vielmehr ist die Kombination von Methoden erforderlich.
- In dieser Handlungssituation sind hohe Anforderungen an die Entscheidungsfindung gestellt. Das erfordert Vorgehensweisen, die es Anwendern und Entwicklern ermöglichen, aus kontroversen Standpunkten gemeinsame Entscheidungen zu machen. Von den genannten Methodenarten unterstützen lediglich die Diskussions- und Besprechungsmethoden diese Prozesse, während sich die übrigen Methodenarten lediglich zur Entscheidungsvorbereitung und zur Informationsbeschaffung eignen.

5.2 Kritische Handlungssituation 2: Auseinandersetzung mit Gestaltungsoptionen

Um Gestaltungsideen und -visionen entwickeln zu können sowie die Möglichkeiten und Grenzen von entwickelten Gestaltungsvorschlägen frühzeitig einschät-

zen zu können, ist die Auseinandersetzung mit Gestaltungsoptionen erforderlich. Diese beinhaltet die Bearbeitung der für den jeweiligen Einsatzzweck in Frage kommenden technischen und organisatorischen Lösungsmöglichkeiten.

- Die Entwickler benötigen einen Überblick über die in Frage kommenden Hard- und Softwareangebote, um Anregungen für Gestaltungen geben zu können und die Grenzen gewählter oder vorgesehener technischer Lösungen einschätzen zu können.
- Die Anwender und die Entwickler eines Projekts benötigen einen Überblick über die realisierbaren organisatorischen Optionen als Orientierung für eigene Gestaltungsvorschläge.

Die Erarbeitung einer zweckgerichteten Beschreibung der technologischen Möglichkeiten erfolgt meist getrennt durch die Entwickler eines Projekts. Auf dem Markt befindliche technische Möglichkeiten werden hinsichtlich ihrer Brauchbarkeit für den konkreten Verwendungszweck bewertet. Dabei sind die gegebenen technischen und finanziellen Voraussetzungen des Projekts ins Verhältnis zu dem Angebot des Marktes zu setzen, um Entscheidungen hinsichtlich der zu wählenden Hard- und Software sowie der zu wählenden Entwicklungsumgebung vorzubereiten. In diesem Zusammenhang muß vorhandene Standardsoftware hinsichtlich ihrer Tauglichkeit für die zur Debatte stehende Projektaufgabe geprüft werden, um Doppelarbeit zu vermeiden. Die aus diesem Prozeß resultierenden Beschreibungen der technischen Optionen bringen die Entwickler ihrerseits in das Projektteam ein.

Die Erarbeitung einer zweckgerichteten Beschreibung der organisatorischen Möglichkeiten erfolgt gemeinsam durch Anwender und Entwickler. Hierbei gilt es, je nach Projektaufgabe alternative Möglichkeiten der organisatorischen Gestaltung einzelner Arbeitsplätze oder abteilungsübergreifender Kooperationsbeziehungen zu bewerten. Für die Erschließung der hierfür erforderlichen Informationen bestehen in den Projekten oft schlechte Voraussetzungen. Während das Wissen über technische Angebote durch die Entwickler oft in das Projekt hineingebracht werden kann, fehlt das Wissen über organisatorische Gestaltungsmöglichkeiten in Software-Projekten meist vollständig. In dem fehlenden Überblick über organisatorische Gestaltungsoptionen liegt häufig der Mangel an Visionen zur Neugestaltung des Arbeitsprozesses begründet. Das fördert die einseitige Konzentration auf die technische Gestaltung und behindert die Suche nach neuen Formen menschengerechter Arbeitsgestaltung.

Aufgrund des oft vollständigen Fehlens entsprechender Erfahrungen sind bei der Erarbeitung der organisatorischen Optionen besondere Maßnahmen zur Informationsbeschaffung notwendig. Wenn das erforderliche Know-how nicht im

Unternehmen vorhanden ist, sollten in dieser Frage externe Experten für die arbeitswissenschaftliche und organisatorische Gestaltung mit herangezogen werden.

Methodisches Vorgehen

Bei der Auseinandersetzung mit **technologischen Gestaltungsmöglichkeiten** besteht das Hauptproblem darin, die erforderlichen Informationen über die technischen Möglichkeiten zu beschaffen und die gefundenen Möglichkeiten in einem geeigneten Verfahren zu bewerten. Die Beschaffung der Informationen ist in erster Linie von der Güte der verfügbaren Informationsquellen abhängig. Die Bewertung der gefundenen Lösungen stellt höhere Anforderungen an das methodische Vorgehen. Sie erfordert einerseits geeignete Kriterien zur rationalen Bewertung der Möglichkeiten und sie erfordert andererseits die Organisation von Entscheidungsverfahren.

Die Auseinandersetzung mit organisatorischen Gestaltungsoptionen ist aufgrund der fehlenden Erfahrungen und Kenntnisse in allererster Linie ein Problem der Informationsbeschaffung und darüber hinaus eine Frage der Informationsbewertung. Bei der Informationsbeschaffung besteht das Problem darin, entsprechende Informationsquellen überhaupt zu finden. Als Informationsquellen können neben dem Studium von einschlägigen Fachzeitschriften insbesondere entsprechende externe Experten zu Rate gezogen werden.

Zur Erschließung der erforderlichen Informationen sowie deren Bewertung können in dieser Handlungssituation insbesondere folgende Methoden eingesetzt werden:

- Dokumentenanalyse,
- Expertengespräche,
- Testverfahren,
- Exkursionen,
- Besprechungs- und Diskussionsmethoden.

5.2.1 Dokumentenanalyse

Die Analyse von Dokumenten ist eine wichtige Informationsquelle zur Erschließung der erforderlichen Informationen über technische und organisatorische Gestaltungsmöglichkeiten. Die Methode stützt sich in dieser Handlungssituation auf die Analyse von

- Fachbüchern und Hintergrundinformationen über fachliche Fragen des konkreten Arbeitsprozesses,
- einschlägige Fachzeitschriften zu technischen und organisatorischen Fragestellungen und
- Herstellerinformationen zu technischen Fragestellungen.

Für die Anwender und Entwickler im Projekt bestehen folgende Einsatzbedingungen für die Dokumentenanalyse in dieser Handlungssituation:

- Die Dokumentenanalyse ermöglicht eine erste Orientierung über in Frage kommende technische und organisatorische Gestaltungsoptionen.
- Durch die Analyse von Fachzeitschriften lassen sich die jeweiligen Entscheidungsalternativen eingrenzen und Parameter für die Entscheidung gewinnen.

Zusammenfassend läßt sich die generelle Eignung der Methode wie folgt einschätzen: Der Einsatz der Dokumentenanalyse eignet sich bei der Erarbeitung einer zweckgerichteten Beschreibung der Gestaltungsoptionen vorwiegend zur Informationsbeschaffung, zur Findung von Entscheidungsalternativen und der Entwicklung geeigneter Bewertungsparameter.

5.2.2 Expertengespräche

Für die Erarbeitung organisatorischer und technischer Optionen der Gestaltung sind Expertengespräche von erstrangiger Bedeutung. Hierbei handelt es sich um Befragungen von Fachleuten zu einem bestimmten Thema (z.B. einer Technologie oder einer organisatorischen Gestaltungsvariante) oder einem ganzen Themengebiet (z.B. einem Marktüberblick über technologische Möglichkeiten oder wissenschaftlich fundierte Kenntnisse über verschiedene Wege der organisatorischen Gestaltung).

Der Einsatz von Expertengesprächen in dieser Handlungssituation eignet sich für folgende Einsatzfälle:

- Expertengespräche ermöglichen eine erste Orientierung über in Frage kommende technische und organisatorische Gestaltungsoptionen.
- Durch Expertengespräche lassen sich die jeweiligen Entscheidungsalternativen eingrenzen und Parameter für die Entscheidung gewinnen.
- Mit Hilfe von Expertengesprächen lassen sich spezielle Kenntnisse und Erfahrungen über Techniken oder organisatorische Gestaltungsvarianten gewinnen, die aufgrund ihrer Aktualität oft noch nicht publiziert sind.

Zusammenfassend läßt sich die generelle Eignung der Methode wie folgt einschätzen: Der Einsatz von Expertengesprächen ist zur Erarbeitung einer zweckgerichteten Beschreibung der Gestaltungsoptionen in größeren Projekten unverzichtbar. Insbesondere notwendige Kenntnisse und Erfahrungen über die organisatorischen Gestaltungsmöglichkeiten lassen sich oft nur auf diesem Weg gewinnen. Dabei dient der Einsatz der Methode vorwiegend zur Informationsbeschaffung, zur Findung von Entscheidungsalternativen und der Entwicklung geeigneter Bewertungsparameter. Die Entscheidungsfindung selbst bedarf entsprechender Prozesse im Projektteam und kann nicht durch das Votum von Experten ersetzt werden.

5.2.3 Testverfahren

Der Einsatz von Testverfahren hat insbesondere bei der Bewertung technischer Gestaltungsmöglichkeiten große Bedeutung, nicht aber für die Bewertung organisatorischer Optionen. Bei der Verwendung von Technologien, mit denen bisher noch keine entsprechenden Erfahrungen gesammelt wurden, muß zunächst bewertet werden, welche Technologie die gesetzten Bedingungen im erforderlichen Umfang erfüllt. Testverfahren werden eingesetzt, um

- gefundene technische Lösungen auf ihre Brauchbarkeit im konkreten Verwendungszweck zu testen,
- auf dem Markt befindliche Standardsoftware auf ihre Tauglichkeit als Basissoftware zu testen,
- die Verträglichkeit unterschiedlicher technischer Produkte untereinander zu überprüfen.

Zusammenfassend läßt sich die generelle Eignung der Methode wie folgt einschätzen: Der Einsatz von Testverfahren ist zur Bewertung technischer Gestaltungsoptionen in den meisten Projekten unverzichtbar. Dabei dient der Einsatz der Methode vorwiegend zur Bewertung der technischen Machbarkeit bestimmter ins Auge gefaßter Lösungen. Die Entscheidungsfindung selbst bedarf aufbauend darauf entsprechender Prozesse im Projektteam und kann nicht durch Testverfahren ersetzt werden.

5.2.4 Exkursionen

Beim Einstieg in eine neue Technologie oder der Erwägung organisatorischer Gestaltungslösungen können Exkursionen in Unternehmen, die diese Gestaltungsoptionen bereits realisiert haben, von großer Bedeutung sein. In der Bewertung von Gestaltungsoptionen sind Exkursionen insbesondere deshalb von

großer Bedeutung, weil sie einen wirklichkeitsnahen Eindruck vermitteln und so für Anwender und Entwickler gleichermaßen eine wichtige Grundlage für die Beschaffung der erforderlichen Informationen sein können. Dabei sind Exkursionen insbesondere aus folgenden Gründen einzusetzen:

- Exkursionen ermöglichen eine erste praxisnahe Orientierung über in Frage kommende technische und organisatorische Gestaltungsoptionen.
- Durch Exkursionen lassen sich die jeweiligen Entscheidungsalternativen eingrenzen und Parameter für die Entscheidung gewinnen.
- Mit Hilfe von Exkursionen lassen sich praktische Erfahrungen über Techniken oder organisatorische Gestaltungsvarianten gewinnen. Insbesondere für die Bewertung vorgesehener Gestaltungsoptionen unter den Bedingungen des »Normalbetriebs« können hier wichtige Informationen gewonnen werden.

Zusammenfassend läßt sich die generelle Eignung der Methode wie folgt einschätzen: Exkursionen bieten Entwicklern und Anwendern im Projekt die Möglichkeit, ins Auge gefaßte technische oder organisatorische Gestaltungsoptionen wirklichkeitsnah zu studieren. Durch diese Konkretion wird die Verständigung des Projektteams untereinander erleichtert und die Kreativität zum Finden eigener Lösungen angeregt. Weiterhin ist die Methode dann unverzichtbar, wenn vorgesehene Lösungen unter »Normalbetrieb« bewertet oder praktische Erfahrungen als Entscheidungsunterstützung beschafft werden sollen.

5.2.5 Besprechungs- und Diskussionsmethoden

Besprechungs- und Diskussionsmethoden sind in dieser Handlungssituation unverzichtbar, um die notwendigen Entscheidungsprozesse über die technischen und organisatorischen Gestaltungsoptionen voranzutreiben. Hat sich das Team erst einmal auf bestimmte technische oder organisatorische Lösungen eingestellt, so beschränkt diese Entscheidung die weitere Suche nach Möglichkeiten der Neugestaltung des Arbeitsprozesses. Somit werden in dieser Handlungssituation wichtige Vorentscheidungen für den weiteren Projektverlauf getroffen, die die Interessen der Projektbeteiligten nachhaltig berühren. Die bisher behandelten Methodenarten dienen dem Team zur Informationsbeschaffung und zur Entscheidungsvorbereitung. Die Entscheidungen selbst sind methodisch durch Besprechungs- und Diskussionsmethoden zu unterstützen.

Besprechungs- und Diskussionsmethoden sind vor allem für folgende Fälle einsetzbar:

- Um zu entscheiden, welche technischen oder organisatorischen Lösungsmöglichkeiten in die engere Betrachtung einbezogen werden;
- um zu entscheiden, welche Kriterien für die Entscheidungsfindung gewählt werden;
- um zu entscheiden, welche organisatorische und technische Gestaltungsoption im konkreten Projekt verfolgt werden soll.
- Darüber hinaus kommen diese Methoden auch in Kombination mit anderen Methoden zur Beschaffung oder Bewertung von Informationen zum Einsatz; gemeinsam mit Testmethoden beispielsweise als Test-Workshops oder kombiniert mit Expertengesprächen als Werkstattgespräch mit Experten.

5.3 Kritische Handlungssituation 3: Entwicklung einer Theorie des zukünftigen Arbeitsprozesses

Der inhaltliche Mittelpunkt des Software-Projekts ist eine Theorie des zukünftigen Arbeitsprozesses. In dieser Theorie werden die wesentlichen Aspekte des zukünftigen Arbeitsprozesses in allgemeiner Form beschrieben. Sie beinhaltet eine Antizipation der zukünftigen Tätigkeiten, der zu gestaltenden Arbeitsmittel und -gegenstände sowie der angestrebten Kooperationsbeziehungen. Diese Handlungssituation unterscheidet sich qualitativ von der zur Entwicklung eines abstrakten Verständnisses der Arbeitswelt durch ihre Hinwendung zum künftigen Realitätsbereich. Wesentlich für diese Situation ist demnach die Antizipation neuer Gebrauchszwecke statt der Rekonstruktion der vorhandenen.

Die Entwicklung einer Theorie des zukünftigen Arbeitsprozesses hat für die Erfüllung der Projektaufgabe folgende Funktionen:

- Sie ist für Anwender und Entwickler gleichermaßen die gültige inhaltliche Beschreibung des angestrebten Arbeitsprozesses. Damit sind die Gestaltungsanforderungen an die Tätigkeiten, die Kooperationsbeziehungen und die Arbeitsmittel und -gegenstände festgelegt.
- Für die Entwickler enthält sie die Anforderungen an die zu erstellenden Software-Produkte.[187]

In der Projektrealität ist die Entwicklung einer Theorie des zukünftigen Arbeitsprozesses zeitlich und inhaltlich eng mit den vorher genannten kritischen Handlungssituationen verbunden. Dies gilt auch dann, wenn sie als eigenständige

[187] Je nach gewählter Entwicklungsumgebung müssen diese Anforderungen in für die technische Umsetzung geeignete Beschreibungsmethoden wie Daten- und Funktionsmodelle umgesetzt werden.

Phase des Projekts abgewickelt wird. Bei der Entwicklung einer Theorie des künftigen Arbeitsprozesses beziehen sich Anwender und Entwickler in einem ständigen Rückbezug auf die abstrakten Beschreibungen des bestehenden Arbeitsprozesses und die konkreten Erfahrungen mit der Arbeitswelt einerseits und die entwickelten technischen und organisatorischen Optionen der Gestaltung andererseits. Das wesentliche Problem dieser Handlungssituation besteht darin, im gemeinsamen Kooperationsprozeß von Anwendern und Entwicklern »Neues« für die Gestaltung des Arbeitsprozesses zu entwickeln und ausreichend genau zu beschreiben.

Zentrale Anforderungen an den Kooperationsprozeß

Traditionelle Methoden des Software Engineering behandeln diese Handlungssituation beispielsweise als »Definitionsphase« [Balzert 1982]. In der Definitionsphase werden die Anforderungen an das Software-Produkt im groben definiert um sie in der Entwurfsphase im Sinne eines software-technischen Konzepts zu realisieren. Dabei wird lediglich der Prozeß der Erstellung des Software-Produkts beachtet. Dessen Entwicklungslogik ist die Grundlage für die Wahl des methodischen Vorgehens. Dementsprechend steht nicht der Prozeß der kooperativen Entwicklung von Gestaltungsanforderungen im Mittelpunkt der Betrachtung; vielmehr ist die Erstellung bestimmter Produkte wie Pflichtenhefte und Funktionshandbücher der Ausgangspunkt für Überlegungen zur methodischen Unterstützung.

Diese Vorgehensweise ist in vielerlei Hinsicht den Anforderungen moderner Software-Projekte nicht mehr angemessen:

- Bei der Software-Entwicklung werden keineswegs nur bereits existierende Tätigkeiten auf den Computer übertragen. Vielmehr wird ein Arbeitssystem mit allen seinen Elementen unter Einsatz von Software neu gestaltet. Der Entwicklungsprozeß beschränkt sich daher nicht nur auf die Definition von Anforderungen an die Software-Produkte. Er mündet vielmehr in die Beschreibung einer Theorie des neuen Arbeitsprozesses. Diese beinhaltet eine Antizipation der zu entwickelnden Software im Zusammenspiel mit den neu zu gestaltenden Tätigkeiten und den veränderten Kooperationsbeziehungen. Das methodische Vorgehen kann sich demnach nicht allein auf die Anforderungen an die Software-Produkte reduzieren, sondern muß den gesamten Wirkungszusammenhang im Arbeitssystem bearbeitbar machen.
- Der Prozeß der Definition von Anforderungen zur Neugestaltung ist keineswegs als linearer Verlauf realisierbar. Der Prozeß der Theorieentwicklung bedarf vielmehr der rückgekoppelten Gestaltung unter Berücksichtigung aller Aspekte des Arbeitsprozesses. Erst wenn beispielsweise die Auswirkungen

avisierter Veränderungen der Software auf die Tätigkeiten und die Kooperationsbeziehungen antizipiert sind, können die Anforderungen an die Software-Produkte definiert werden. Daraus ergibt sich die Notwendigkeit eines evolutionären Entwicklungsprozesses. In diesem Prozeß gelingt es Anwendern und Entwicklern erst nach und nach durch das Aufeinanderzubewegen der unterschiedlichen Kompetenzen die erforderliche Gestaltungskompetenz gemeinsam zu erwerben.

- Der Prozeß der Definition von Gestaltungsanforderungen und der der Realisierung lassen sich in der Realität moderner Software-Entwicklung nicht trennen. Aufgrund der Komplexität der Wirkungsverhältnisse zwischen den einzelnen Aspekten des Arbeitsprozesses ergibt sich die Notwendigkeit, die Entwicklung einer Theorie des zukünftigen Arbeitsprozesses als dialektischen Prozeß von Aneignung und Vergegenständlichung zu organisieren und entsprechend methodisch zu unterstützen.

Methodisches Vorgehen

Das Vorgehen in der kritischen Handlungssituation zur Entwicklung einer gemeinsamen Theorie hat verschiedene Anforderungen zu berücksichtigen. Die Wahl der Methoden ist demnach übergreifend nicht an der Definition von Anforderungen an das Software-Produkt zu orientieren. Vielmehr ist das Vorgehen so anzulegen, daß sich Anwender und Entwickler in gemeinsamer Kooperation auf eine gemeinsame Theorie des zukünftigen Arbeitsprozesses verständigen. Das erfordert vor allem ein evolutionäres Vorgehen bei beständiger Beachtung der Wechselwirkungen zwischen den einzelnen Elementen des Arbeitsprozesses und ein Vorgehen, das die sukzessive Entwicklung von Gestaltungsanforderungen im Zusammenspiel von Aneignung und Vergegenständlichung möglich macht.

Für die Entwicklung einer Theorie des zukünftigen Arbeitsprozesses können folgende Methoden eingesetzt werden:

- Besprechungs- und Diskussionsmethoden,
- Prototyping,
- Testverfahren.

5.3.1 Besprechungs- und Diskussionsmethoden

Besprechungs- und Diskussionsmethoden sind im Prozeß der Entwicklung einer Theorie unverzichtbar. Nur durch den Einsatz dieser Methoden lassen sich die erforderlichen Kooperationsprozesse von Anwendern und Entwicklern in dieser

Handlungssituation koordinieren. Dabei hat der Einsatz dieser Methoden folgende Ziele:

- Besprechungs- und Diskussionsmethoden werden eingesetzt, um Visionen der Neugestaltung des Arbeitsprozesses zu entwickeln und zu bewerten.
- Besprechungs- und Diskussionsmethoden werden eingesetzt, um vorgeschlagene Gestaltungslösungen zu bewerten.
- Besprechungs- und Diskussionsmethoden werden eingesetzt, um Entscheidungen über Gestaltungslösungen herbeizuführen.

Für die Entwicklung von Visionen zur Neugestaltung und Eingrenzung der zu verfolgenden Lösungsvorschläge eignen sich Workshops oder Zukunftswerkstätten. Bei Zukunftswerkstätten wird in einem Dreischritt von Kritik des Bestehenden, Entwicklung von Visionen und Selektion der Problembereiche die Leitorientierung der Gestaltung gemeinsam von Anwendern und Entwicklern entwickelt. In diese Sitzung fließen die Erfahrungen der Anwender mit dem bestehenden Arbeitssystem sowie die entwickelten Optionen zur Gestaltung ein. Über die Bearbeitung von Visionen des neuen Arbeitsprozesses werden die strategischen Entwicklungslinien des Projekts zwischen den Projektbeteiligten abgestimmt. Anwender und Entwickler sind in diesen Sitzungen gleichermaßen aktiv beteiligt. Die Entwickler haben darüber hinaus meist die Aufgabe der Vorbereitung, Moderation und Dokumentation der Sitzung.

Besprechungsmethoden können weiterhin eingesetzt werden, um die verschiedenen Folgewirkungen angestrebter Lösungen zu bewerten. Hierbei werden Gestaltungsvorschläge in gemeinsamen Gesprächen hinsichtlich ihrer Auswirkungen auf die Tätigkeiten, die Kooperationsbeziehungen und die Software evaluiert. Wenn die Gestaltungsvorschläge bereits sehr konkret ausgearbeitet sind, werden die Sitzungen durch verschiedene Testverfahren ergänzt.

Die Verabschiedung der Theorie des zukünftigen Arbeitsprozesses erfolgt ebenfalls über Besprechungs- und Diskussionsmethoden. Hierbei kommt es darauf an, daß der Entscheidungscharakter dieser Sitzungen durch entsprechende Methoden unterstützt wird.

Besprechungs- und Diskussionsmethoden sind die wesentliche methodische Unterstützung für die erfolgreiche Durchführung der Entscheidungsprozesse des Projektteams. Sie sind in dieser Handlungssituation weiterhin von großer Bedeutung für die Erzeugung von Visionen zur Neugestaltung und von konkreten Gestaltungsvorschlägen.

5.3.2 Prototyping

Verfahren des Einsatzes von Prototypen haben eine hervorgehobene Rolle bei der Entwicklung einer Theorie des künftigen Arbeitsprozesses. Sie werden insbesondere eingesetzt, um Vorschläge zur Gestaltung zu sammeln und entwickelte Vorstellungen zu veranschaulichen und zu bewerten. Der Einsatz von Prototypen eignet sich für die Unterstützung der Kooperation von Anwendern und Entwicklern insbesondere, weil die Kommunikations- und Lernprozesse zwischen beiden Gruppen gezielter auf die jeweils zur Debatte stehende Problemstellung bezogen werden können und weil der Umgang mit Prototypen eine konkrete, sinnliche Erfahrung des jeweiligen Gestaltungsvorschlags vermittelt. Desweiteren ist durch den Einsatz von Prototypen die kontinuierliche Qualifizierung der Anwender während des Projektes möglich, weil sie über die Weiterentwicklung der Prototypen beständig an der Weiterentwicklung der Software beteiligt sind. [Mehl u.a. 1993] Je nach Einsatzzweck sind folgende Verfahren des Prototyping[188] zu unterscheiden:

Exploratives Prototyping: Dieses Verfahren dient dazu, die Anforderungen der Anwender an das zu erstellende Software-Produkt zu besprechen. Dabei wird das neuzugestaltende Software-Produkt in Form von statischen Bildern der Benutzungsoberfläche dargestellt und anhand typischer Arbeitsabläufe mit den Anwendern besprochen. Aufbauend darauf werden je nach Vorgehen im Projekt die Anforderungsspezifikation als Teil des Pflichtenheftes oder erste lauffähige Prototypen entwickelt.

Experimentelles Prototyping: Im Rahmen dieses Verfahrens werden die Prototypen mit den Anwendern nicht nur besprochen, sondern auch erprobt. Durch den praktischen Umgang mit den Prototypen werden bei Anwendern und Entwicklern Lernprozesse unterstützt, um die Vorstellungen zur Neugestaltung des Systems präziser formulieren zu können. Floyd u.a., die dieses Verfahren im Projekt "PEtS" zur Unterstützung der Kooperation von Anwendern und Entwicklern angewandt haben, unterscheiden folgende Varianten: komplette Funktionssimulation, Simulation der Dialogschnittstelle, Skelett-Programmierung (Realisierung einiger, besonders wichtig erscheinender Systemkomponenten) und Konstruktion von Basisfunktionen. [Mehl u.a. 1993; Floyd u.a. 1994]

---

[188] Seit dem Erscheinen des Sammelbandes "Approaches to prototyping" [Budde u.a. 1984] hat sich die Anzahl der Neuerscheinungen zum Thema »Prototyping« stark vergrößert. Einen Überblick über das Thema bieten: Budde u.a. [1992] und Holz auf der Heide [1993].

Evolutionäres Prototyping: Bei der Entwicklung komplexer Systeme wird dieses Verfahren angewandt, um das System nach und nach aus einem Teil heraus zu entwickeln. Der letzte Prototyp entspricht dann dem Endprodukt.

Verfahren des Prototyping sind für die methodische Unterstützung der Kooperation von Anwendern und Entwicklern zur Entwicklung einer Theorie des künftigen Arbeitsprozesses - zumindest bei umfangreicheren Problemstellungen - unverzichtbar.[189] Sie unterstützen die Verständigung zwischen beiden Gruppen und die Entscheidungsfindung dadurch, daß sie den zur Debatte stehenden Gestaltungsvorschlag veranschaulichen. Sie erleichtern den Theoriebildungsprozeß im Wechselverhältnis von Theorie und praktischer Umsetzung und fördern so die Herausbildung von Gestaltungskompetenz bei Anwendern und Entwicklern.

5.3.3 Testverfahren

Der Einsatz von Testverfahren ist in den meisten Projekten bereits in der Handlungssituation der Theoriebildung unverzichtbar. Testverfahren dienen dazu, die Brauchbarkeit einer vorgesehenen Lösung frühzeitig zu bewerten. Durch die Ergebnisse der Tests gewinnt das Projektteam wichtige Anhaltspunkte zur weiteren Entwicklung von Gestaltungsvorschlägen. Der Einsatz von Testverfahren zur Unterstützung des Prozesses der Entwicklung einer Theorie des künftigen Arbeitsprozesses kann sich sowohl auf Simulationen des künftigen Systems - in Form von schriftsprachlichen Beschreibungen oder Zeichnungen - beziehen, als auch auf rechnergestützte Darstellungen der Gestaltungsvorschläge in Form von Prototypen. Insbesondere in Kombination mit dem Einsatz von Prototypen zur Entwicklung der Gestaltungsanforderungen haben Testverfahren eine wichtige Bedeutung.

Bei der Bewertung von Gestaltungsvorschlägen anhand von Prototypen[190] sind folgende Vorgehensweisen besonders wichtig [Holz auf der Heide 1993]:

Expertenbasierte Methoden: Experten bewerten die Brauchbarkeit auf der Basis wissenschaftlicher Grundlagen zur menschengerechten Gestaltung von Software. Ein Beispiel für diese Methode ist das Verfahren EVADIS-II [Oppermann u.a. 1992], das dazu verwandt wird, die Benutzerschnittstelle auf ihre software-ergonomische Qualität zu überprüfen.

---

[189] Bezüglich der Voraussetzungen des Einsatzes von Prototypen für die Entwicklungsumgebung schildert Holz auf der Heide [1993] die Erfahrungen aus dem Forschungsprojekt.

[190] Sehr ausführlich und konkret wird die Verwendung von Testmethoden zur Bewertung von Prototypen bei Holz auf der Heide [1993] geschildert. Hier findet sich auch ein ausgezeichneter Literaturüberblick.

Benutzerbasierte Verfahren: Die Benutzer bewerten den Systemvorschlag anhand ihrer Erfahrungen. Dieses Verfahren wird mit anderen Methoden wie Befragungen und Beobachtungen kombiniert und gilt als zentrales Bewertungsverfahren. [Holz auf der Heide 1993]

Versuchspersonenbasierte Verfahren: Die Arbeit von Versuchspersonen mit dem entsprechenden Prototyp wird bewertet. Dieses Verfahren basiert nicht auf der Einschätzung der Anwender. Testergebnisse werden vielmehr durch die Bewertung der Aufgabenerfüllung durch die Versuchsperson gewonnen. Diese Testmethode wird häufig in Kombination mit Beobachtungen - insbesondere Videoaufzeichnungen - oder Rechnerprotokollen verwandt.

Testverfahren sind eine wesentliche Grundlage, um vorgesehene Gestaltungslösungen frühzeitig zu bewerten. Damit gewinnen Anwender und Entwickler im Prozeß der Theoriebildung wichtige Anhaltspunkte für die schrittweise Weiterentwicklung von Gestaltungsvorschlägen. Testverfahren eignen sich als Grundlage und Hilfe für die notwendigen Entscheidungsprozesse im Projekt. Sie sind aber kein Ersatz für die gemeinsame Entscheidungsfindung im Team.

6 Zusammenfassung

Moderne Software-Entwicklung ist als kooperative Gestaltung zu verstehen. Diese beinhaltet gemeinsame Lernprozesse zwischen Entwicklern und Anwendern, das Lösen von Konflikten und übergreifend die Herstellung einer erfolgreichen Kommunikationsbeziehung. Nicht das Erzeugen von Software-Produkten bestimmt in erster Linie den Entwicklungsprozeß, sondern die Dynamik der Kooperationsprozesse von Anwendern und Entwicklern.

Die methodische Handhabung der Kooperationsbeziehungen zwischen Anwendern und Entwicklern in Software-Projekten läßt sich anhand vier **kritischer Handlungssituationen** thematisieren. Diese sind:

1. Die Erarbeitung eines Verständnisses des Arbeitsprozesses,
2. die Auseinandersetzung mit Gestaltungsoptionen,
3. die Entwicklung einer Theorie des zukünftigen Arbeitsprozesses und
4. die Implementierung des entwickelten Software-Produkts.

Für jede dieser kritischen Handlungssituationen lassen sich die jeweiligen Schwerpunkte der Kooperationsbeziehung zwischen Anwender und Entwickler, die erforderlichen Produkte und die Anforderungen an die methodische Unterstützung diskutieren. Erst das Verständnis der Besonderheiten der jeweiligen kritischen Handlungssituation ermöglicht den Entwicklern einen

handlungskompetenten Umgang mit den Methoden. Bei der Vermittlung von Kooperationskompetenz an die IT-Fachkräfte sollte die problemorientierte Bearbeitung dieser kritischen Handlungssituationen im Mittelpunkt stehen.

V Prozeßsteuerung in der Software-Entwicklung

Der Prozeß der Software-Entwicklung wurde bisher in seinen Dimensionen Arbeitsgestaltung und Kooperation dargestellt. Beide Dimensionen wurden analytisch aus dem Betrachtungsgegenstand Software-Entwicklung herausgelöst und hinsichtlich ihrer inhaltlichen Bestimmungsmomente betrachtet. Im Software-Projekt sind diese Aspekte nicht zu trennen. Hier verschmelzen Arbeitsgestaltung und Kooperation zu einer Einheit und müssen gemeinsam bewältigt werden.

Mit diesen beiden Momenten moderner Software-Entwicklung ist eine Zunahme der Anforderungen an die aufgabenangemessene Steuerung des Entwicklungsprozesses zu verzeichnen, die nicht mehr allein in das Projektmanagement verlagert werden kann, sondern in Teilen zu den Aufgaben aller Teammitglieder gehört. Prozeßkompetenz gehört damit zu den zentralen Qualifikationsanforderungen an IT-Fachkräfte. Sie muß in ihrem Aufgabenverständnis verankert und operationalisiert werden, damit eine entsprechende Handlungskompetenz herausgebildet werden kann.

Mit diesem Kapitel wird das Wissen über den Entwicklungsprozeß in seinen Bestandteilen »Arbeitsgestaltung« und »Kooperation« unter der Problemstellung »Prozeßsteuerung« verdichtet. Ausgehend von allgemeinen Überlegungen zur Notwendigkeit der Planung und Steuerung des Entwicklungsprozesses werden verschiedene Formen der Prozeßsteuerung diskutiert und damit einer situationsbezogenen Operationalisierung zugänglich gemacht.

1 Die Steuerung des Software-Entwicklungsprozesses

1.1 Die Notwendigkeit der Prozeßsteuerung

Software-Projekte müssen gesteuert werden. Sie sind heute in ihrem Verlauf so kompliziert und intransparent, daß sie geradewegs ins Chaos laufen und eine Projektruine hinterlassen, wenn sie nicht durch steuernde Maßnahmen flankiert werden. Die Ursachen dieser Komplexität sind vielfältig. Der Umfang und das Abstraktionsniveau moderner Softwaresysteme sind enorm groß. Sie können durch den einzelnen Menschen intellektuell kaum noch vollständig erfaßt werden. Ohne eine Strukturierung des Software-Prozesses, die sich in einer durchschaubaren Struktur des Programmcodes niederschlägt, wird die anwachsende

Menge der »lines of code« für die Projektbeteiligten schnell undurchschaubar. Hinzu kommt der steigende Abstraktionsgrad der Programme, die nicht mehr allein mit dem »gesunden Menschenverstand« nachzuvollziehen sind.

Aber nicht nur die Software selbst trägt zur Komplexität des Projekts bei. Software-Entwicklung ist immer Teamarbeit. Die Notwendigkeit von Arbeitsteilung und Kooperation in großen Projekten führt zu einer Vielzahl von parallel oder zeitversetzt verlaufenden Teilaktivitäten, die aufeinander und auf die Gesamtaufgabe des Projektteams bezogen werden müssen. Hinzu kommt, daß nicht nur innerhalb des Projektteams, sondern auch außerhalb verschiedene Gruppen mit dem Projekt befaßt sind, die ebenfalls koordiniert werden müssen. So sind etwa Geschäftsleitungen, Anwender, Betriebsräte, Rechenzentrumsmitarbeiter und Systemlieferanten mit den Aufgaben des Software-Projekts befaßt.

Software-Entwicklung ist gleichermaßen eine kreative und kooperative Arbeit, der weitgehend eine immanente strukturierende Form fehlt. Damit ist ihr eine **Tendenz zur Chaotik** beigegeben, die im Projekt in produktiven Grenzen gehalten werden muß. Darüber hinaus stellen Praktiker eine psychologische Notwendigkeit der Prozeßsteuerung fest: "(...) es ist weder für Entwickler psychisch erträglich noch für den Auftraggeber akzeptabel, allein das Endergebnis - das laufende System - als Erfolgskriterium zu haben. Alle brauchen Zwischenergebnisse, an denen sie ablesen können, ob und wie sie vorankommen." [Denert 1992, S. 49]

Die **Prozeßsteuerung** erfüllt im Software-Projekt damit eine zentrale Funktion. Mit ihr werden

- die intellektuelle und organisatorische Bewältigung des Umfangs und des Abstraktionsgrades der Software,
- die Koordination von Einzelarbeiten und -prozessen,
- die Kontrollierbarkeit des Ressourceneinsatzes und der Zielerreichung,
- die Motivation der Teammitglieder und
- die Zufriedenheit der Auftraggeber über Ziel/Weg-Festlegungen und die Definition von Zwischenergebnissen

angestrebt.

1.2 Anforderungen an die Prozeßsteuerung

Das Verständnis von Prozeßsteuerung im Software-Projekt ist heute vor allem durch das **Software Engineering** und seine Grundannahmen geprägt. Software-Entwicklung wird als Konstruktionsprozeß begriffen, der ausschließlich der

Herstellung von Software-Produkten dient. Die idealisierenden Grundannahmen, die dieser Disziplin zugrunde liegen, faßt Floyd so zusammen: "Softwareentwicklung beruht auf vorgegebenen Problemen mit fest definierbaren Anforderungen; der Herstellungsprozeß ist anhand von Prozeßmodellen formalisierbar; der Einsatzkontext von Software kann ausgeklammert werden." [Floyd 1994, S. 29] Mit dem Software Engineering wurde eine **für den Konstruktionsprozeß weitgehend durchgängige Methodik** entwickelt, die die Modellierung, den Software-Entwurf und die Programmierung unterstützt. Abgeleitet wurde eine an Konstruktionsschritten orientierte Prozeßstruktur, die die Grundlage vieler Software-Projekte wurde. Aus historischen Gründen priorisiert das traditionelle Software Engineering mit seinen Grundannahmen zwei Projektanforderungen: Sie ist zum einen an den Erfordernissen technischer Anwendungen ausgerichtet, die in der Phase der Entwicklung des Software Engineering in den 70er Jahren einen Boom erlebten. Zum anderen entspricht es tayloristischen Rationalisierungsvorstellungen, die bisher die Unternehmensstrategien prägten und auf einer Formalisierung von Arbeit basierten. [ebd.]

Solchermaßen geprägt, war in dieser Disziplin für andere Aspekte und Probleme der Software-Entwicklung lange Zeit keinen Raum. So blieben etwa das **Verhältnis von formaler und informaler Welt sowie die sozialen Anforderungen** im Software-Projekt weitgehend ausgeblendet. Erst seit wenigen Jahren wird über eine Innovation des Software Engineering nachgedacht. Sie wird die mit der Zunahme nicht-technischer Anwendungen und der Ablösung tayloristischer Unternehmensstrategien notwendig wird, denn den Besonderheiten moderner Software-Projekte sind die idealisierenden Vorstellungen des traditionellen Software Engineering nicht mehr angemessen. Es geht mit seiner Konzentration auf den Konstruktionsprozeß der Software von **Annahmen** aus, die für moderne Software-Projekte nur noch sehr eingeschränkt Gültigkeit besitzen und keinesfalls mehr als allgemeingültige Leitbilder für Software-Entwicklung taugen:

- "Software dient zur Lösung wohldefinierter, formalisierbarer Probleme. Gegenstand der Softwareentwicklung ist die Herstellung eines Produkts.
- Die Herstellung von Software kann von ihrem Einsatz getrennt werden, die Herstellung basiert auf festen Anforderungen, die vorweg ermittelt werden.
- Die Herstellung des Produktes besteht aus Konzeption und Realisierung, gefolgt von Wartung. Der Herstellungsprozeß kann anhand von vordefinierten Phasen mit vordefinierten Zwischenergebnissen in Form von Dokumenten (Meilensteinen) gegliedert und standardisiert werden.

- Die Konzeption besteht in der Erarbeitung eines formalen Modells, das in einer Spezifikation ausgedrückt wird. Die Spezifikation legt fest, was das Programm leistet, nicht wie es auf dem Basissystem realisiert wird.
- Die Realisierung besteht in der Umsetzung der Spezifikation in ein Programm oder ein Programmsystem durch sukzessive Schritte der Verfeinerung und Transformation.
- Die Arbeitsteilung kann anhand der Produktstruktur frühzeitig und durchgehend erfolgen. Die Kommunikation wird über Dokumente geregelt." [ebd., S. 30]

Mit diesen Annahmen hat das traditionelle Software Engineering eine **Klasse von Programmierproblemen** vor Augen, die relativ einfach strukturiert sind. Eine Klassifikation von Programmierproblemen finden wir bei Lehmann. [Lehmann 1980] Lehmann interpretiert Software als eine Abbildung von Realitätsausschnitten. Er untersucht das Problem der Wechselwirkungen zwischen Abbildung und der resultierenden Veränderung der Realität und identifiziert drei Problemklassen:

S-Programme sind exakt spezifizierbar. Die exakte Abbildung der Realität in der Spezifikation und der Spezifikation im Programm ist beweisbar. Beispiele sind viele mathematische Probleme wie der kleinste gemeinsame Teiler.

P-Programme sind entweder nicht exakt spezifizierbare oder nicht exakt implementierbare Programme. Das Programm muß pragmatisch bestimmt werden.

E-Programme sind nicht exakt spezifizierbare Programme, die in einem Rückkopplungsprozeß auf die Realität einwirken und damit die eigenen Ausführungsbedingungen verändern. In diese Klasse fällt die nicht-technische Anwendungssoftware. [ebd.]

Das Software Engineering geht von exakt spezifizierbaren und programmierbaren Programmen aus. In der Konsequenz müssen alle Programmierprobleme, die P-Programmen oder E-Programmen entsprechen, vereinfacht werden, um diesem Schema zu entsprechen. Bei einigen Problemen war und ist diese **Vereinfachung** pragmatisch sinnvoll und erfolgreich. Moderne Software-Projekte erfordern allerdings eine **Berücksichtigung des Einsatzkontexts der Software** in einer Form, die sich dieser Vereinfachung entzieht. Die Annahmen vordefinierter Prozesse, gegebener, klar strukturierter Probleme und einer Ausblendung des Kontextes werden hier zu behindernden Restriktionen.

Moderne Software- und Hardware-Architekturen zeichnen sich durch Miniaturisierung und Objektorientierung aus. Immer kleinere Einheiten mit zunehmend einfacher Bedienbarkeit kennzeichnen diese Entwicklung, deren erklärtes Ziel es

ist, den Computer in allen Lebens- und Aufgabenbereichen einsetzen zu können. Sie führt zu einer gegenseitigen Durchdringung von Software und sozialer Realität, die über einfache Schnittstellenmuster nicht mehr zu erklären und zu gestalten ist. Die **Wechselwirkungen zwischen der Softwareentwicklung und der Veränderung des Arbeitsprozesses** sind in Folge nicht mehr vorab bestimmbar, sondern führen zu unscharfen Zielformulierungen, die keine Formalisierung der Probleme und des Prozesses zulassen. Statt dessen müssen die Wechselwirkungen von Software und Arbeit im Prozeß iterativ erkannt und angemessen berücksichtigt werden. Damit wird Arbeitsgestaltung neben der Software-Konstruktion zur zweiten Aufgabe des Software-Projekts. Sein Verlauf ist durch das Verhältnis dieser beiden Aufgaben bestimmt und nicht mehr auf den technischen Konstruktionsprozeß reduzierbar.

Der **Prozeß der Software Entwicklung** wird hierdurch außerordentlich komplex und nur schwer durchschaubar, so daß Aussagen über seinen tatsächlichen Verlauf nur sehr schwer zu treffen sind. Dagegen findet er aber meist unter so einschränkenden ökonomischen Bedingungen hinsichtlich Zeit, Kosten und Qualitätsanforderungen statt, daß die **Anforderungen** an seine Vorhersehbarkeit und Planung sehr hoch sind. Gefordert sind in der Regel Aussagen,

- die Verlauf und Qualität des Prozesses nachvollziehbar und überprüfbar machen und so das Risiko eines Scheitern des Projekt verringern.
- die die Erzeugung und Qualität des Endprodukts vorab bestimmen und hier ebenfalls Fehlinvestitionen möglichst ausschließen.

Notwendig ist eine Prozeßsteuerung, die auf begründeten Annahmen und Festlegungen hinsichtlich des Projektverlaufs basiert und eine weitestmögliche Transparenz, Planbarkeit und Kontrollierbarkeit des Prozesses ermöglicht.

Die zweckgerichtete **Prozeßsteuerung** der Softwareentwicklung muß dabei zumindest folgende Aspekte der Software Entwicklung umfassen, um diesen Anforderungen weitestmöglich zu genügen:

- das Kooperationsverhältnis von Anwendern und Entwicklern und ihre Aufgabenteilung sowie
- das Verhältnis von Arbeitsgestaltung und Software-Realisierung.

2 Produktorientierte Ansätze der Prozeßsteuerung

2.1 Modelle als Planungsgrundlage

Die als ideal angesehene Vorgehensweise wird häufig in **Vorgehensmodellen** abgebildet, die eine zentrale Planungsgrundlage von Projekten bilden. Diese Vorgehensmodelle, die einigen Autoren auch als Projektmodelle bezeichnen, werden im allgemeinen für eine geordnete Projektabwicklung für unerläßlich gehalten. Denert nennt folgende Bestandteile, die für Manager und Entwickler eine gemeinsame und verbindliche Sicht der logischen und zeitlichen Struktur eines Projekts definieren sollen:

- "die grundlegenden Prinzipien und Begriffe,
- die Phasen, in denen ein Projekt abläuft,
- die darin vorgesehenen Aktivitäten und die damit zu erzielenden Ergebnisse,
- die anzuwendenden Methoden, Techniken und Richtlinien
- sowie die sie unterstützenden Werkzeuge und Hilfsmittel,
- die Meilensteine und die Qualitätssicherungs- (QS-) Maßnahmen, mit denen das Erreichen der Meilensteine festgestellt wird." [Denert 1992, S. 32]

Die Phasen und Aktivitäten dieser Vorgehensmodelle basieren auf den Projektbestandteilen:

- Projektetablierung
- Anforderungsanalyse
- Realisierung
- Nutzung/Test

Diese grundlegenden Aktivitäten und ihre Produkte (wie Pflichtenheft und Software) werden im Vorgehensmodell in eine zeitliche und inhaltliche Reihenfolge gebracht, an der der reale Projektverlauf orientiert wird.

In vielen Unternehmen werden die geltenden, oft sehr formalistischen Projektmodelle in umfangreichen **Entwicklungshandbüchern und Richtlinienbänden** dokumentiert. Denert weist zu recht auf die Gefahr hin, daß diese Projektmodelle - vorausgesetzt sie werden aktiv genutzt und verstauben nicht nur im Schrank - eine starke **»Software-Bürokratie«** begründen und kreative Arbeit im Projekt ersticken.

Ihrer Form nach ähneln Projektmodelle in der Tat sehr stark einer Methodik, die für alle Projekte gleichermaßen anwendbar sein soll. Unter der Überschrift

"Projektmodelle versus Schablonen-Methodik" wehrt sich DeMarco vehement gegen diese Ansicht und vertritt die These, daß Projektmodelle projektspezifisch zu entwickeln und zu handhaben seien.

"In dem Maße, wie ich älter und wackliger geworden bin, ertappe ich mich dabei, mehr und mehr über die unflexiblen Regelwerke verbittert zu sein. Viele Firmen versuchen, diese Regelwerke in eine Methodik zu pressen, die dann als verbindliches Heiligtum gepriesen wird. Die Idee jedoch, daß eine einzelne Methodik in zwei verschiedenen Projekten vorherrschen soll, ist bereits äußerst verdächtig: Die Unterschiede zwischen verschiedenen Projekten sind wesentlich wichtiger als die Ähnlichkeiten." [DeMarco 1989, S. 185]

DeMarco schlägt in Auswertung seiner langjährigen Projektleitertätigkeit vor, im Projektteam ein spezifisches Projektmodell zu entwickeln und einzusetzen. Dies kann sinnvoll auf einer vorgegebenen Methodik beruhen, die allerdings angepaßt werden sollte. Für dieses Vorgehen sieht er eindeutige Effektivitätsvorteile, da sich das Projektteam selbsterarbeitete Pläne schneller aneignet und in der Regel nicht unbeachtet läßt, wie dies bei vorgegebenen Modellen häufig der Fall ist.

2.2 Das Wasserfallmodell

Das Vorgehensmodell, das den meisten anderen in Gebrauch befindlichen phasenorientierten Modellen zugrunde liegt, ist das **Wasserfallmodell**. Das Wasserfallmodell, das Boehm zugeschrieben wird, wurde in der Praxis und in der Literatur immer wieder aufgegriffen und verändert. [vgl. Boehm 1981] Es kursieren einige dieser Modelle mit einer unterschiedlichen Anzahl und Bezeichnung der Phasen. Eine sehr allgemeine Definition finden wir bei Raasch. [Raasch 1991, S. 412] Er unterscheidet folgende **Phasen**:

- Voruntersuchung,
- Analyse,
- Design,
- Realisierung,
- Abnahme,
- Einführung.

Anzahl und Bezeichnung der Phasen sind davon abhängig, welche Produkte im Software-Entwicklungsprozeß als notwendig erachtet werden. Für diese Produkten können die Aufgaben des Software-Projekts sehr unterschiedlich zusammengefaßt sein. Unternehmen, die entsprechende Richtlinien entwickeln, und Autoren, die Modelle vorschlagen, können dabei zu recht unterschiedlichen Er-

gebnissen kommen, die entweder auf Erfahrungen mit sehr unterschiedlichen Projekttypen oder einfach auf der Kenntnis und dem bevorzugten Einsatz bestimmter Methodensets beruhen, die einen bestimmten Verlauf erfordern.

Denert etwa beschreibt folgende **Dokumente**:

- die Systemspezifikation,
- die Systemkonstruktion,
- die Modul-Programmierung und
- die Systemintegration[191]. [vgl. Denert 1992, S. 83ff.]

Die **Systemspezifikation** ist jenes Dokument, das die Sicht des Anwenders auf das System definiert. Im wesentlichen besteht sie aus drei Teilen:

- "Das *Datenmodell* ist die Grundlage der Spezifikation. Es vermittelt eine konzeptionelle, anwendungsbezogene Darstellung der Daten, mit denen das zu spezifizierende System arbeitet.
- Das *Funktionenmodell* legt die Funktionen des Systems fest und verknüpft sie mit dem Datenmodell. Eine wichtige Rolle spielt dabei das *Zustandsmodell*, das die Lebensläufe bestimmter Datenobjekte widerspiegelt.
- Die *Benutzerschnittstelle* tritt im Dialog und als Batch in Erscheinung und legt die Art der Datenrepräsentation für und der Funktionenbenutzung durch den menschlichen Benutzer fest. Die *Nachbarsysteme-Schnittstelle* bestimmt Kommunikation und Datenaustausch mit anderen Systemen." [ebd. S. 40]

Die **Systemkonstruktion** legt die softwaretechnische Konzeption in drei Teilen fest:

- Mit der *Modularisierung* wird das zu entwerfende System in Teile/Module zerlegt und das grobe Zusammenwirken der Module bestimmt.
- Mit dem *Datenbasisentwurf* wird die Organisation der Daten auf den externen Speichermedien festgelegt, die etwa durch ein Datenbankverwaltungssystem verwaltet werden.
- Mit der *Prozeßorganisation* wird der Ablauf von Modulen und die Kommunikation von Modulen über Prozeßgrenzen hinweg bestimmt.

---

[191] Denert ist kein Verfechter eines streng phasenorientierten Vorgehens. Wir verwenden hier lediglich seine übersichtliche Darstellung der von ihm vorgeschlagenen Dokumente im Entwicklungsprozeß.

Für die Module muß jeweils eine **Modulprogrammierung** durchgeführt werden. Sie besteht aus

- der *Modulspezifikation*, die die Funktionalität des Moduls beschreibt und seine Schnittstellen definiert,
- der *Modulkonstruktion*, also der Erzeugung von Programmcode und
- dem *Modultest* in einem Testsystem.

Mit der **Systemintegration** werden die Module zu einem laufenden System zusammengeführt. Ergebnisse können

- eine *Subsystemdefinition* bei größeren Systemen,
- ein *Subsystemtest*,
- der *Systemtest* sein.

Die Dokumentation entsteht projektbegleitend aus den Teilen, die keinen Programmcode enthalten. Sie besteht aus der Systemspezifikation, der Systemkonstruktion, den Modulspezifikationen, den Modulkonstruktionsbeschreibungen, den Subsystemdefinitionen und den Testfällen.

Unabhängig davon, welche Phasen und Dokumente definiert werden, sind den Wasserfall-Modellen folgende **Grundannahmen** gemeinsam:

- Das Vorgehen ist dokumentenorientiert. In jeder Phase entsteht ein Dokument, das vollständig abgeschlossen und möglichst vom Auftraggeber abgenommen wird. Die folgende Phase beginnt erst nach Abschluß der vorhergehenden. Geringe, wenn auch nicht erwünschte Rückkopplungsmöglichkeiten gibt es ausschließlich mit der unmittelbar vorangegangenen Phase.
- Die Phasenergebnisse bilden die Dokumentation der Software. Damit ist die Vorstellung verbunden, daß die Dokumentation redundanzfrei ist und mit dem Programm gewartet wird.
- Der Benutzer soll zu Beginn des Projekts die Anforderungen an die Software vollständig und konsistent definieren. Die Anforderungen verändern sich im Projektverlauf nicht.
- Die prozeßbegleitende Qualitätssicherung basiert auf den Phasenergebnissen als Meilensteinen. [vgl. Spitta 1989, S. 26f.]

Der **Vorteil** des Wasserfallmodells liegt in seiner Einfachheit begründet. Es ist einfach strukturiert und für den Auftraggeber und das Projektteam unmittelbar transparent.

Für Raasch hat das Wasserfallmodell dennoch gravierende **Nachteile**:

- "Reale Projekte folgen nicht unbedingt der sequentiellen Abfolge der Phasen. Iterationen treten häufig auf und sind in der Planung nicht leicht abzubilden.
- Reale Projekte halten nur selten die strenge Trennung zwischen je zwei Phasen ein.
- Das Produkt ist erst nützlich anwendbar, wenn es vollkommen fertiggestellt ist. Weder der Anwender noch das Management kann sich ein Bild von der Qualität des Systems machen, bevor es fertig ist. Der Anwender hat eventuell keine Chance, sich graduell an die neue Arbeitsweise mit dem System zu gewöhnen.
- Obwohl das Projekt fieberhaft am Ergebnis arbeitet und obwohl ständig Dokumente erzeugt werden, hat man kein zuverlässiges Kriterium für den tatsächlichen Projektstand. Es entsteht ein Sicherheitsgefühl, weil permanent Dokumente erzeugt werden. Deren Qualität kann aber meistens nicht abschließend beurteilt werden. Man erkennt trotz aller fertiggestellten Dokumente nicht die Größenordnung des Berges, der noch vor einem liegt.
- Meilensteine sind als Mittel der Qualitätssicherung untauglich. Statt dessen werden Hilfen zur begleitenden Qualitätssicherung benötigt.
- Vom Anwender wird erwartet, daß er in der ersten Projektphase alle Anforderungen explizit artikulieren kann. Mit der zu Projektanfang meistens vorliegenden allgemeinen Unsicherheit wird das Wasserfallmodell nicht leicht fertig." [Raasch 1991, S. 413]

Spitta kritisiert zusätzlich

- die Annahme eines statischen Benutzers, dessen Anforderungen sich nicht verändern und
- die Gleichbehandlung aller Dokumente hinsichtlich ihrer Bedeutung in der Dokumentation. "Dies führt in der Praxis regelmäßig dazu, daß die hochgradig redundante Dokumentation überhaupt nicht gewartet wird." [Spitta 1989, S. 27]

Wir heben folgende Probleme hervor, die die Einsatzmöglichkeiten des Wasserfallmodells erheblich einschränken:

a) Das phasenorientierte Vorgehen erzeugt eine **Intransparenz und Bürokratisierung des Prozesses**, die sich motivations- und kreativitätshemmend für das Projektteam auswirken. Für einfache und gut strukturierte Probleme kann das Wasserfallmodell eine gute Grundlage für die Entwicklung eines

spezifischen Projektmodells darstellen. In der Regel aber zeichnen sich Software-Projekte eben gerade nicht durch diese Merkmale aus. Sie erscheinen zunächst als ein nicht zu bewältigender Berg, den sich das Projektteam Schritt für Schritt erarbeiten muß. Das Wasserfall-Modell zwingt dazu, Problem- und Anforderungsdefinitionen in den ersten Phasen des Projekts umfassend und erschöpfend vorzunehmen, bevor die nächsten Phasen begonnen werden können. Da dieses Vorgehen i.d.R. auf große Probleme (z.B. mangelnde Zielkonstanz im Anwendungsgebiet) trifft, bleiben für das Projektteam zwei Möglichkeiten:

- Entweder wird das Problem stark vereinfacht. Es werden Anforderungen definiert, die nur noch begrenzten Aussagewert für das tatsächlich zu lösende Problem haben. In diesem Falle kann zwar mit den weiteren Konstruktionsschritten begonnen werden. Das Projektteam befindet sich aber mit dieser Entscheidung in einem **illusionären Qualitätsverständnis**, das spätestens in der Phase der Implementierung und des Tests durch den Anwender aufgedeckt wird.
- Die zweite Möglichkeit besteht darin, **viel Zeit und Aufwand in die Phase der Anforderungsdefinition zu stecken**. Da es an tragfähigen Entscheidungskriterien fehlt, wird definiert und spezifiziert, es werden Unmengen nicht mehr überschaubarer Dokumente produziert. Aus Zeitdruck wird diese Phase dann irgendwann willkürlich abgebrochen; man wird feststellen, daß in der Masse nicht die Klasse steckt.

Das Projektteam schiebt dabei eine Menge von Dokumenten vor sich her, die keinerlei Aussagen über den realen Stand des Projektes mehr zulassen. Entscheidungen können nur noch bürokratisch, nach den Richtlinien des vorgegebenen Vorgehensmodell getroffen werden. Für die Teammitglieder gilt als einziger Orientierungspunkt für die Bewertung der eigenen Leistungen die Umsetzung der herausgegebenen Richtlinien und Konventionen. Motivation und Kreativität bleiben hier auf der Strecke.

b) Durch den vorgeschriebenen relativ frühen Abschluß der Anforderungsdefinition wird ausgeschlossen,

- daß der **Anwender die formulierten Anforderungen im Projektverlauf weiter konkretisiert und an die tatsächlich bestehenden Anforderungen annähert** und
- daß das Software-Projekt **in umfassende Innovationsprozesse eingebunden wird**, die über den Projektverlauf hinweg zu einer Neugestaltung des Anwendungsbereichs und damit zu einer Veränderung der Anforderungen führen.

Ersteres geht von der unschlüssigen Annahme aus, der Anwender könne das Anwendungsfeld vollständig rational durchdringen, mit den Gestaltungsmöglichkeiten moderner Informationstechnik verknüpfen und aus diesem Wissen Anforderungen ableiten. Die Realität in den Projekten aber sieht anders aus: Anwender sind in Bezug auf ihr Arbeitsfeld mit einer Betriebsblindheit behaftet, die es ihnen unmöglich macht, alle Aspekte des Anwendungsfeldes zu erkennen und zu formulieren. Diese Betriebsblindheit ist allerdings nichts ungewöhnliches oder ein Effekt, der nur bei technisch unversierten Anwendern eintritt. Jeder wird die Erfahrung machen, daß die exakte Beschreibung gerade alltäglicher und häufig ausgeführter Tätigkeiten allergrößte Schwierigkeiten bereitet. Man denke nur an eine formale Beschreibung der Vorbereitung des alltäglichen Frühstücks. In der Beschreibung dieser Routinetätigkeiten wird wohl niemand alle relevanten Details beachten. Informatiker selbst können dieses Phänomen an sich selbst feststellen, wenn sie an die letzte Situation denken, in der sie Anwendern eine Software erklären wollten, die sie selbst »im Schlaf« beherrschen. Je intensiver die Auseinandersetzung mit der zukünftigen Software ist und je mehr Anschauungsmaterial zur Verfügung steht (etwa in Form von Prototypen), um so mehr werden die Anforderungen konkretisiert und erweitert werden. Werden diese in die Realisierung des Produkts nicht aufgenommen, ist die **Unzufriedenheit der Anwender** mit der Software vorprogrammiert. Viele Projekte verlagern in Reaktion auf diese Unzufriedenheit die Umsetzung dieser Anforderungen in die **Wartung**. Die Konsequenz ist, daß die Wartung teurer wird als die Entwicklung, ohne daß das gewünschte Ergebnis im nachhinein noch erbracht werden könnte.

Der zweite Aspekt gewinnt seit einigen Jahren an Bedeutung. Die **Annahme einer weitgehend starren Arbeitsorganisation im Anwendungsfeld** trifft auf diejenigen Unternehmen nicht mehr zu, die Software-Projekte im Rahmen umfassender Innovationsprojekte ansiedeln. Die Einführung integrierter Sachbearbeitung, Geschäftsprozeßorientierung, die Einführung von Gruppenarbeit u.a. können das Anwendungsfeld im Projektzeitraum grundsätzlich verändern. Die Orientierung am Wasserfall-Modell hieße, die Beteiligung des Software-Projekts an diesen Innovationen grundsätzlich zu verneinen und damit die Suche nach geeigneten Vorgehensweisen aufzugeben.

2.3 Statische Prozeßsteuerung über Meilensteine

Im Rahmen **produktorientierter Phasenkonzepte** wird der Projektverlauf über Phasen und deren jeweilige Zwischenprodukte gesteuert. Bestimmte Teilergebnisse des Projekts werden als **Etappenziele** festgelegt, die als **»Meilensteine«**

bezeichnet werden. "Ein Meilenstein gilt als erreicht, wenn das betreffende Ergebnis fertig ist und gewissen Qualitätsstandards genügt. Dies festzustellen ist Sache einer Qualitätssicherungs-(QS-) Maßnahme. Für jeden als wichtig erachteten Meilenstein wird ein Soll-Termin geplant und der Ist-Termin registriert, zu dem der Meilenstein tatsächlich erreicht wird, das Ergebnis also wirklich fertig ist." [Denert 1992, S. 49] Wichtige Meilensteine ergeben sich aus den Spezifikationen und den Tests bzw. Testergebnissen. Maßnahmen der Qualitätssicherung sind v.a. Reviews und Inspektionen, also die Begutachtung der Zwischenprodukte durch ein Team oder eine einzelne Testperson. [ebd.]

Einschränkend bemerkt Denert: "Man muß und sollte nicht jeden mögliche Meilenstein mit Soll/Ist-Terminen planen bzw. kontrollieren, so wenig wie sich der Bergsteiger für jeden markanten Punkt in der Landschaft eine Zeit vorgibt oder der Marathon-Läufer an jedem seiner 42 km die Zwischenzeit nimmt. Man darf Planung und Kontrolle eben auch nicht übertreiben!" [ebd.]

Denert spricht hier ein zentrales **Problem der Projektsteuerung über Meilensteine** an: die Tendenz zu einer übermäßigen Formalisierung und Kontrolle des Projektverlaufs, die schnell zu einer »Software-Bürokratie« führen kann. DeMarco skizziert diese Form der Projektsteuerung in ihrer extremen Form so: "Heutzutage gibt es Dutzende von Projektkontrollsystemen auf dem Markt. Sie unterscheiden sich hauptsächlich in der Form und Quantität der Datenausgaben und versuchen, sich gegenseitig in der Aufmachung der Ergebnisberichte und im purem Umfang des Datenausstoßes zu übertreffen. Sie arbeiten alle mehr oder weniger auf die folgende Art und Weise:

1. Sie unterteilen Ihr Projekt in so viele Aufgaben wie nur möglich. Das Projektkontrollsystem wird Ihnen wahrscheinlich Aufgabenstellungen vorschlagen, die durch Checklisten erschöpfend beschrieben sind und möglicherweise eine maximale Größe für Aufgabenstellungen festlegen.
2. Sie holen eine detaillierte Schätzung der für jede Aufgabenstellung benötigten Manpower ein. Wie Sie das machen, wird vom Projektkontrollsystem höchstwahrscheinlich nicht kommentiert.
3. Sie vergeben für jede Aufgabenstellung eine eigene Kontrollnummer.
4. Ihre Projektmitarbeiter legen die Zeitdauer für die einzelnen Aufgabenstellungen in Zahlen fest.
5. Wenn die für eine bestimmte Aufgabenstellung bemessene Zeit aufgebraucht ist, wird der Job als erledigt betrachtet.

Natürlich ist die im letzten Punkt gemachte Annahme kompletter Unsinn (...)." [DeMarco 1989, S. 193f.]

Die Projektsteuerung über Meilensteine erzeugt eine **Illusion von Kontrollierbarkeit und Steuerbarkeit**, die in vielen Projekten zu diesen bürokratischen Auswüchsen führte. Sie basiert auf den fehlerhaften Annahmen, daß

- Zwischenergebnisse in großer Anzahl zu Beginn des Projekts definiert und
- diese Zwischenergebnisse trotz der notwendigerweise unscharfen Aufgabenformulierung mit einem quantifizierbaren Aufwand bewertet werden können.

Daraus resultieren zwei Kernprobleme: einerseits begibt sich das Projekt mit dieser Vorgehensweise in ein **Korsett vordefinierter Schritte und Zwischenergebnisse**, die sich mit dem Fortschreiten des Projekts immer weiter von den situationsabhängigen Notwendigkeiten der Entwicklung entfernen. Es werden in Folge Aufgaben bearbeitet, deren Sinn zunehmend zweifelhaft wird; andere wichtigere Aufgaben müssen aufgrund mangelnder Kapazitäten zurückgestellt werden. Diese Form der Projektsteuerung bedeutet im allgemeinen das Aus für Kreativität und die effektive Kooperation zwischen Anwendern und Entwicklern.

Andererseits entsteht mit der Illusion der Aufwandskontrolle die paradoxe Situation, daß **reale Arbeitsstände nicht mehr realistisch eingeschätzt werden können**. DeMarco beschreibt: "Die Aufgabenstellungen sind oftmals so nebulös formuliert, daß fast jede Art von Aufwand zugeteilt werden kann, ohne daß irgendjemand in seiner Leichtgläubigkeit etwas davon bemerkt.

Das Ergebnis von alledem ist, daß nur wenige Jobs Zeitüberschreitungen zeigen, bis fast alle Aufgabenstellungen ihre zugeteilte Zeit aufgebraucht haben (...)." [ebd.] Das Resultat besteht letztlich darin, daß angebliche 90% der Aufgaben (oder mehr) bis zur geplanten Auslieferung fertiggestellt sind, die Fertigstellung der angeblichen restlichen 10% aber zu einer Verzögerung der tatsächlichen Auslieferung um mehrere 100% führt. Viele Projekte sind schon seit Monaten oder sogar Jahren »fast fertig«.

3 Prozeßorientierte Ansätze der Prozeßsteuerung

Die Probleme und Unzulänglichkeiten der traditionellen Ansätze der Prozeßsteuerung im Software-Projekt haben dazu geführt, daß neue, moderner Software-Entwicklung angemessenere Vorgehensweisen entwickelt wurden, mit denen der **nicht-lineare Verlauf des Prozesses,** das **rückgekoppelte Verhältnis von Arbeitsgestaltung und Software-Entwicklung** und **die Zusammenarbeit von Entwicklern und Anwendern** besser unterstützt werden können. Es wurden neue modellhafte Vorstellungen hinsichtlich des Verlaufs von Software-Projekten entwickelt und die Produkte des Software-Projekts hinsichtlich ihrer

Entstehung und ihrer Bedeutung im Spannungsfeld zwischen Software-Konstruktion und Arbeitsgestaltung neu bestimmt.

Grundlegende Ansätze der Projektsteuerung sind

- zyklische Vorgehensmodelle in der Projektplanung,
- Referenzlinien statt Meilensteine in der situativen Projektsteuerung,
- Theoriebildung zur Koordinierung der Prozesse der Arbeitsgestaltung und Software-Konstruktion sowie
- Prototyping zur Unterstützung und Steuerung erfahrungsorientierter Lernprozesse.

Gemeinsam ist diesen Ansätzen, daß sie die statische, produktiorientierte Sicht auf den Entwicklungsprozeß aufgeben und statt dessen die prozeßorientierten, dynamischen Elemente der Prozeßsteuerung hervorheben. Der Notwendigkeit von Rückkopplungsmöglichkeiten soll Rechnung getragen und der Raum situationsspezifischer und variabler Handlungsmöglichkeiten soll vergrößert werden.

3.1 Das Projektmodell STEPS

Moderne Ansätze der Prozeßsteuerung kennzeichnen den Prozeß der Software-Entwicklung treffend als **evolutionär**. Arbeitsprozeß und Software nehmen Schritt für Schritt Gestalt an, indem einzelne grundlegende Aktivitäten des Projekts immer wieder durchlaufen und die Produkte immer wieder überarbeitet werden. Zur Steuerung dieser Prozesse werden seit Mitte der achtziger Jahre verstärkt **zyklische Vorgehensmodelle** diskutiert und eingesetzt, die die Rückkopplung zwischen Anforderungsdefinition und Software-Realisierung thematisieren.[192] Dieses Modelle basieren nicht mehr auf der Prämisse, daß bereits zu Beginn des Projekts eine vollständige Anforderungsdefinition möglich sein muß. Statt dessen ist auf der Basis von Prototypen ein zyklischer Verlauf vorgesehen, der eine iterative Bearbeitung der Produkte (v.a. der Software) und die Rück-

[192] Im Laufe der achtziger Jahre ist in der wissenschaftlichen Diskussion eine Erosion des Phasenmodells zu beobachten. Etappen dieser Entwicklung sind der Ansatz einer "prozeßorientierten Systementwicklung", den Floyd bereits 1981 vorstellte, das Labyrinthmodell von Liedtke [Liedtke 1985] und das Konzept der "evolutionären Systementwicklung" von Budde u.a. [Budde u.a. 1986]. Nach Coy [Coy 1992] läßt sich in der zweiten Hälfte der achtziger Jahre ein grundlegender Wandel des Verständnisses vom Software-Engineering beobachten. Vetreter dieser neuen Richtung sind Nygaard [Nygaard 1986], Winograd/Flores [Winograd/Flores 1989], Floyd [Floyd 1987] und Ehn [Ehn 1988]. Dieser Wandel - weg vom "produktorientierten", hin zum "prozeßorientierten" Methodenverständnis [Floyd] - macht sich insbesondere hinsichtlich der geforderten Projektmodelle bemerkbar.

kopplung zwischen verschiedenen Zwischenergebnissen (der Spezifizierung und der Realisierung) ermöglicht..

Viele der heute verwendeten zyklischen Projektmodelle gehen auf Floyd und ihren Ansatz STEPS zurück. STEPS steht für "Softwaretechnik für Evolutionäre und Partizipative Systementwicklung". [Floyd u.a. 1989]

Die Software-Entwicklung wird als Folge von Entwicklungszyklen angesehen, wobei jeder Zyklus die Erstellung und den Einsatz einer lauffähigen **Systemversion** zum Gegenstand hat. [Floyd u.a. 1994, S. 33] Die Aktivitäten der Projektbeteiligten im Entwicklungsprozeß lassen sich »idealisierend« in vier zentrale Teilprozesse zusammenfassen:

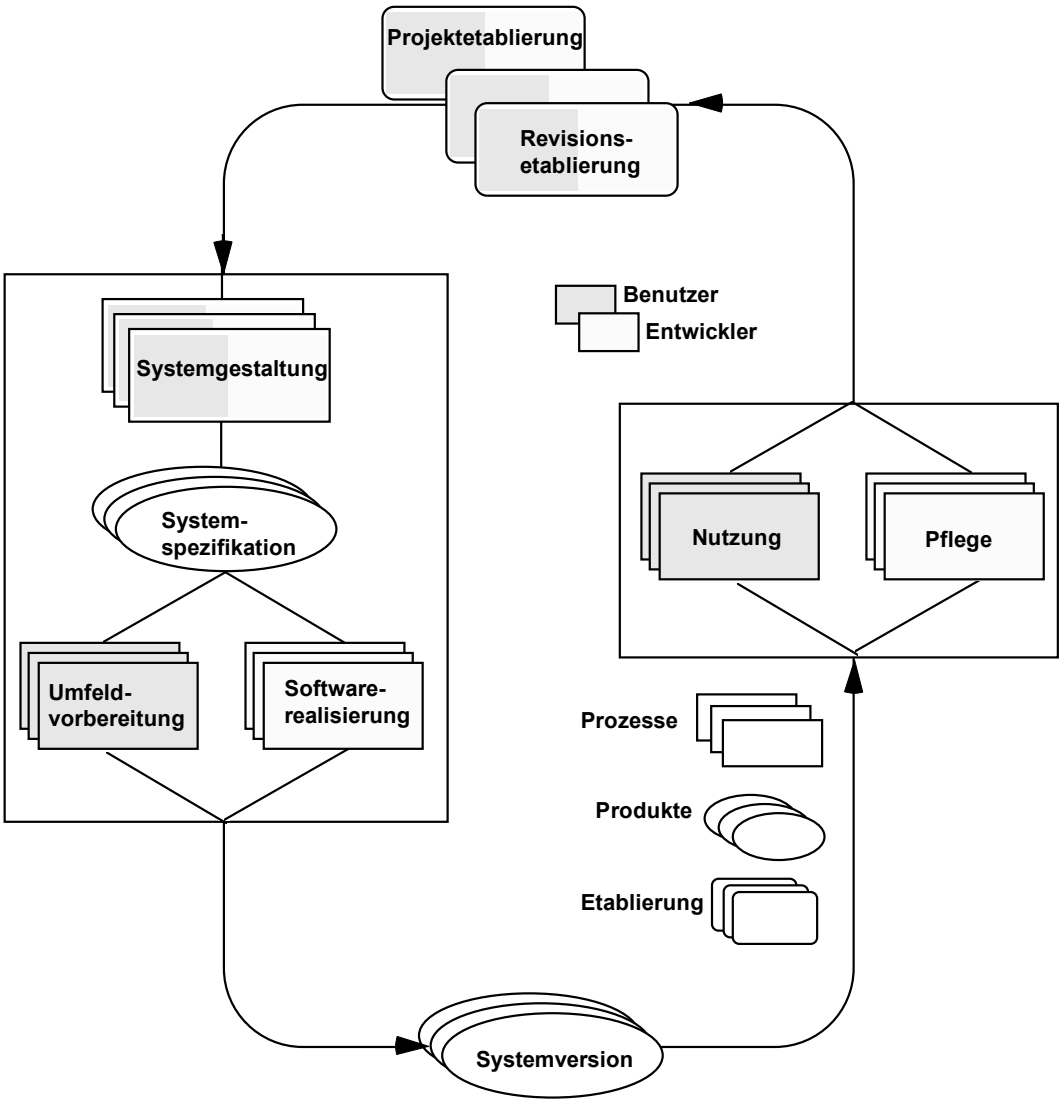
- Die **kooperative Etablierung eines Projektzyklus**. Hier werden die Ziele des nächsten Entwicklungsschritts, die Rahmenbedingungen des Entwicklungsprozesses und die Vorgehensweise vereinbart. An diesem Abstimmungsprozeß sind nicht nur die Projektmitglieder beteiligt, sondern auch »mittelbar Beteiligte« wie Auftraggeber und Betriebsrat.[193]
- Die **Gestaltung** beinhaltet alle Tätigkeiten der Ermittlung, Bewertung und Festlegung der Eigenschaften der Software und der zukünftigen Arbeit. Sie erfolgt in Kooperation von Entwicklern und Anwendern.
- Die **Realisierung** beinhaltet die Tätigkeiten der Definition, des Entwurfs sowie der Implementierung und Validierung des Softwaresystems. Sie erfolgt in Kooperation der Entwickler untereinander.
- Die **Nutzung** beinhaltet alle Tätigkeiten des Gebrauchs der Software-Produkte durch die Anwender. [ebd., S. 35]

In Abhängigkeit von den spezifischen Bedingungen des Entwicklungsprojekts werden diese Teilprozesse im Verlauf der jeweiligen Entwicklungszyklen in jeweils eigener Form und Dynamik vollzogen. Am Beispiel des ersten Entwicklungszyklus lassen sich die Produkte und Aktivitäten schematisch wie folgt darstellen (vgl. Abbildung).

---

[193] Hinsichtlich der Bedeutung der Zyklusetablierung führen die Autoren aus: "Der Kerngedanke der Zyklusetablierung ist, daß grundlegende Konflikte zwischen den am Projekt unmittelbar und mittelbar Beteiligten ausgeräumt sein müssen, damit die eigentliche Entwicklung des Systems als kooperativer Arbeitsprozeß initiiert werden kann." [Floyd u.a. 1994, S. 41]

Projektmodell STEPS



Quelle: [ebd., S. 33]

Zyklusetablierung

Den Beginn eines jeden Entwicklungszyklus bildet die **Zyklusetablierung**. Dieser wird eine besondere Bedeutung zugemessen, da ein kooperativer Gestaltungsprozeß nur auf der Grundlage kooperativ vereinbarter Ziele erfolgreich bewältigt werden kann. [ebd., S. 41] Die Etablierung des ersten Zyklus wird als Projektetablierung vollzogen, deren Ergebnis ein Kooperationsvertrag sein soll.

Die Initiierung der Folgezyklen wird als Revisionsetablierung bezeichnet. Hier werden die jeweiligen Revisionserfordernisse für die Überarbeitung der Softwareversion festgelegt. [ebd., S. 37f.]

Systemgestaltungsarbeiten

Die **Systemgestaltungsarbeiten** bilden das Kernstück der kooperativen Gestaltungsprozesse von Entwicklern und Anwendern. Wesentlich für die Systemgestaltung ist, daß in den kooperativen Gestaltungsprozessen von Anwendern und Entwicklern nicht nur die Eigenschaften des Systems bestimmt , sondern auch das hierfür benötigte Wissen und die zur Bewertung der Qualität erforderliche gemeinsame Kompetenz in wechselseitigen Lernprozessen gebildet werden. Erst in dieser doppelten Entwicklung "erhält das System seine Konturen und mit zunehmender gemeinsamer Gestaltungskompetenz gewinnt es nach und nach seine stabile Gestalt." [ebd., S. 38f.]

Folgende Aufgaben stehen im Zentrum der Systemgestaltungsarbeiten:

- die Ermittlung der Anforderung,
- die Gestaltung der Benutzung,
  - Bestimmung der Qualitäts- und Strukturmerkmale,
  - Arbeitsorganisation, Arbeitstätigkeiten und Rechnereinbettung,
  - Funktionalität und Handhabung,
  - Organisationsstrukturen und Kooperationsbeziehungen,
- Festlegung der Systemfunktionen,
- Entwurf und ggf. Realisierung des Datenmodells,
- Entwurf der Benutzermaschine,
  - Entwurf der Dialogschnittstelle und Simulationen von Dialogen,
- Bestimmung der Basismaschine,
  - anzuschaffende oder bereitzustellende Hard- und Softwarekomponenten. [ebd., S. 38]

Softwarerealisierung

Auf der Grundlage der Systemspezifikation wird die **Software** in der Zusammenarbeit der Entwickler realisiert. Das Ergebnis dieser Arbeiten ist die Systemversion. Diese umfaßt neben dem Einsatzrechner und dem Software-Produkt alle Dokumente, die zur Erstellung der Version von Entwicklern und Anwendern erarbeitet wurden und als Vorgaben für die Benutzung und Pflege des Systems dienen. Die Arbeitsschritte sind hier:

- die Erstellung des Systementwurfs, also der "internen" Architektur des Software-Produkts,

- die Implementierung und der Test einzelner Systemkomponenten,
- die Funktions- und Leistungsüberprüfung der arbeitsteilig programmierten Komponenten im Zuge der schrittweisen Integration zu einer vollständigen Version. [ebd., S. 39]

Nutzung und Versionspflege

Die Version des lauffähigen Software-Produkts wird in jedem Zyklus unter regulären Arbeitsbedingungen von den Anwendern genutzt. Währenddessen sind die Entwickler mit der Versionspflege, z.B. Fehlerbehebung und Optimierung, beschäftigt. [ebd.]

Nach diesen Arbeiten der Nutzung und Versionspflege wird gegebenenfalls ein weiterer Entwicklungszyklus etabliert. Dabei richtet sich die Anzahl der Bearbeitungszyklen nach den konkreten Erfordernissen des jeweiligen Projekts. Der Abschluß der Arbeiten in nur einem Zyklus wird von den Autoren als "Spezialfall" angesehen. [ebd., S. 44]

Zyklische Projektmodelle, für die mit STEPS ein dokumentiertes Beispiel vorliegt, sind eine **notwendige Grundlage der prozeßorientierten Steuerung** moderner Software-Projekte. Ihre besondere Bedeutung ist darin begründet, daß sie **Handlungsräume eröffnen und strukturieren**, in denen sich Kooperation und Kreativität entfalten können. Im Gegensatz zu den produktorientierten Ansätzen bieten sie den Raum für notwendige Rückkopplungen

- zwischen Software-Entwicklung und Arbeitsgestaltung sowie
- zwischen Software-Konstruktion und Anforderungsdefinition.

Darüber hinaus können mit ihnen individuelle Lernprozesse von Anwendern und Entwicklern berücksichtigt und systematisch auf die angestrebten Ergebnisse des Projekts bezogen werden. Ein zyklisches Modell ist dabei als Methodenrahmen zu verstehen, der einen projektspezifischen Einsatz von Methoden unterstützt. Sinnvoll ergänzt werden zyklische Projektmodelle durch weitere Ansätze der prozeßorientierten Steuerung, die eher situative und produktbezogene Aspekte berücksichtigen.

3.2 Dynamische Prozeßsteuerung durch Referenzlinien

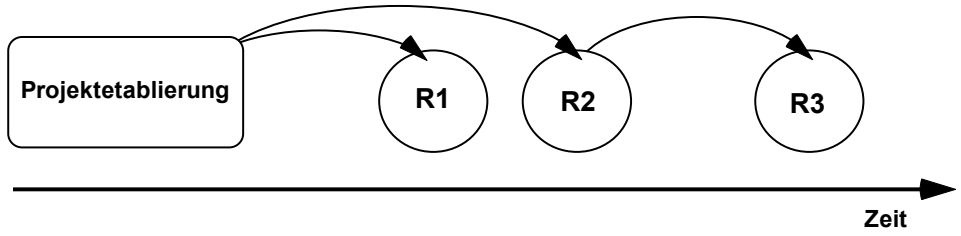
Das Konzept der Referenzlinien, das auf Andersen [Andersen u.a. 1986] zurückgeht, korrespondiert mit dem Einsatz zyklischer Projektmodelle. Es wurde als **Alternative zur Steuerung über Meilensteine** entwickelt und ermöglicht einerseits **variables, situationsabhängiges Handeln**, bildet andererseits aber

dennoch einen Planungshorizont, an dem sich das Projektteam und externe Beteiligte orientieren können.

Es wird davon ausgegangen, daß bei Projektbeginn nur wenige Meilensteine sinnvoll bestimmbar sind. Somit können diese Meilensteine auch nur geringen Einfluß auf die Prozeßsteuerung ausüben. Zwischen diesen Meilensteinen müssen Teilprozesse und Tätigkeiten dynamisch, d.h. situationsspezifisch koordiniert werden.

Durch eine situationsspezifische Definition von Zwischenprodukten als **Referenzlinien** sollen die Tätigkeiten und Teilprozesse des Projekts auf die jeweils situationsbedingt notwendigen Erkenntnis- und Gestaltungsfortschritte ausgerichtet werden. Entsprechend definiert Reisin den Begriff »Referenzlinie« als "einen Projektzustand, den die Entwickler und Benutzer aus dem Entwicklungsprozeß heraus zur Synchronisation ihrer jeweiligen unterschiedlichen und gemeinsamen Arbeitsprozesse vereinbaren. Eine Referenzlinie wird vereinbart, indem ein angestrebter Projektzustand durch die Spezifikation eines (Ziel-) Produkts (oder eine Liste aus mehreren Produkten), auf das die Arbeitsprozesse der Entwickler und Benutzer gerichtet werden sollen, definiert wird und die Kriterien benannt werden, die es erfüllen muß, bevor der nächste Projektzustand in Angriff genommen werden kann." [Reisin 1992, S. 70] Form und Zeitpunkt dieser Zwischenprodukte sind im Gegensatz zu Meilensteinen nicht durch das Projektmodell vorgegeben, sondern werden prozeßbezogen definiert. Anhand eines Beispiels soll die Steuerung des Projektverlaufs über Referenzlinien vedeutlicht werden:

Koordinierung eines Entwicklungsprozesses durch Referenzlinien



Quelle: [Reisin 1992, S. 71]

In einer Projektetablierung wird ein Projektvertrag (PV) vereinbart. Dieser enthält eine Vereinbarung über zwei Referenzlinien (R 1 und R 2), die beispielsweise ein Modell zur Veranschaulichung neugestalteter Arbeitsabläufe und eine Bildschirmrepräsentation der Benutzungsschnittstelle beinhalten können. Ein Teil der

Vereinbarung bezüglich Referenzlinie 2 ist die Festlegung, daß hier weitere Referenzlinien zu vereinbaren sind.

Dabei besteht eine Referenzlinie aus

- einem oder mehreren Zwischenprodukten (z.B. eine zu installierende Systemversion, ein auszuwertender Prototyp, ein bestimmtes Dokument),
- Kriterien zur Bewertung der Zwischenprodukte,
- Bewertungsverfahren und -zuständigkeit,
- Kriterien zur Abnahme,
- Verfahren zur Abnahme und Abnahmezuständigkeiten,
- Durchführung der Referenzlinie. [Mehl u.a. 1993, S. 42]

Art und Inhalt der Referenzlinien wird von den jeweiligen Besonderheiten des Projekts bestimmt. Referenzlinien dienen der Koordination der Teilprozesse und Tätigkeiten innerhalb des Projekts und sind inhaltlich, zeitlich und methodisch an diesen Koordinationserfordernissen auszurichten. Wesentlich für die Definition von Referenzlinien sind

- die inhaltlichen Erfordernisse der Gestaltung von Arbeit und Software und
- die dynamischen Erfordernisse des kooperativen Lernprozesses.

Indem Referenzlinien immer **in Abhängigkeit realer Projektzustände** und Entwicklungsmöglichkeiten definiert werden, können sie einen überschaubaren Planungshorizont schaffen, der für alle Projektbeteiligten transparent und kontrollierbar ist. Notwendige Rückkopplungsschleifen und die Revision veralteter Ergebnisse werden durch dieses Vorgehen unterstützt. Mit der damit verbundenen Reduktion einengender formaler Regeln und unangemessener Aufgabenzuteilungen kann sich die Kooperation zwischen Anwendern und Entwicklern und die Kreativität im Team entwickeln.

3.3 Produkte des Software-Projekts zwischen Software-Konstruktion und Arbeitsgestaltung

Mit dem Konzept der Referenzlinien wird die Anzahl fest definierter **Produkte** des Software-Projekts tendenziell reduziert. Das Software-Projekt kann sich auf zwei zentrale Produkte konzentrieren:

a) die Software und

b) ein Produkt, das die Merkmale der Software systematisch in die Vorstellungen über den zukünftigen Arbeitsprozeß einbettet. Dieses Produkt bezeichnen wir im folgenden als »Theorie des Arbeitsprozesses«.

Das Hauptaugenmerk in der Betrachtung und Darstellung dieser Produkte liegt weniger auf ihrer inneren Struktur als vielmehr auf ihrer Entstehung.

3.3.1 Theoriebildung

Mit der wachsenden Bedeutung der Arbeitsgestaltung in Software-Projekten wird die alleinige Konzentration auf das technische Produkt Software (incl. der Nebenprodukte wie Dokumentationen) zunehmend in Frage gestellt. Die Software allein ist nicht mehr hinreichend, um die angestrebte Optimierung des Arbeitsprozesses zu verwirklichen. Notwendig ist ein weiteres Produkt, das den **Arbeitsprozeß als ganzen in seiner Einheit von Tätigkeiten, Arbeitsmitteln und -gegenständen sowie der Kooperation** zum Gegenstand hat.

In Verbindung mit Ansätzen, die Software-Entwicklung als »Design« verstehen, wurde der Begriff der **»Theoriebildung«** in Anlehnung an den Ansatz des "theory building view" nach Naur geprägt. [Naur 1985]

"Softwareentwicklung wird nicht mehr ausschließlich als Prozeß der systematischen Analyse einer gegebenen Realität und schrittweisen Formalisierung analysierter Anforderungen begriffen. In den Vordergrund rücken nunmehr Arbeitstätigkeiten, die eher als explorativ und experimentell bestimmt sind und die als reflektorische und diskursive Aktivitäten, als Theoriebildung, aufgefaßt werden." [Reisin 1992, S. 82]

Die Theoriebildung in der Software-Entwicklung ist dabei kein Akt individueller Genialität, sondern vorrangig als gemeinsamer Lernprozeß zu betrachten, in dem ein neuer Arbeitsprozeß schrittweise projektiert wird. Dieser findet zwischen den am Projekt beteiligten Entwicklern und Anwendern statt. "Entscheidend sind Kommunikations- und wechselseitige Lernprozesse zwischen den Entwicklern und Benutzern, in deren Verlauf gewissermaßen ein neuer Realitätsbereich geschaffen wird. Mit dem Wissen, das in ihrem Verlauf hervorgebracht wird, gewinnt das System seine Konturen und mit der wachsenden gemeinsamen Gestaltungskompetenz erhält es nach und nach seine stabile Gestalt." [ebd., S. 83].

Um den diesen Einsatz zu ermöglichen, muß die Theorie als schriftsprachliches Produkt entwickelt werden. In der Theoriebildung ist eine übergeordnete Synchronisation und Koordination der Aktivitäten notwendig, die die Kontinuität des Entwicklungsfortschritts für alle Beteiligten nachvollziebar macht. Hierzu sind

Darstellungsmittel erforderlich, die eine kontinuierliche, stabile und für alle Projektbeteiligten verständliche Symbolisierung des angesammelten Gestaltungswissens erlaubt. Theoriebildung muß deshalb die Herstellung schriftsprachlicher Produkte beinhalten, die den Stand symbolisieren, Entwicklung vermitteln und über den unmittelbaren Entwicklungszusammenhang hinaus verständlich sind. Von Methoden ist deshalb zu fordern, daß sie den Aufbau einer Theorie für alle Beteiligten nachvollziehbar und kontrollierbar machen. Die Beschreibungen müssen dabei nicht alle Aspekte umfassen, wesentlich ist, daß sie konsensfähig sind.

Neben ihrer Bedeutung im kooperativen Lernprozeß von Anwendern und Entwicklern kommt der Theoriebildung zentrale Bedeutung in der **Vermittlung von Arbeitsgestaltung und Softwareentwicklung** zu.

Die Wechselwirkungen zwischen Arbeit und Software sind nicht mehr punktuell zu erfassen. Sie befinden sich in einem reziproken Verhältnis, in dem kleinste Veränderungen des einen die Ausführungsbedingungen des anderen grundlegend verändern können. Arbeit und Software müssen aus einer ganzheitlichen Sicht heraus gestaltet werden, welche Software als einen integralen Bestandteil des Arbeitsprozesses begreift.

Die konsequente Verfolgung der Arbeitsgestaltung über den gesamten Projektverlauf hinweg bietet für das Software-Projekt die Möglichkeit, die mit der Software-Entwicklung verfolgten Ziele der Effektivierung des Arbeitsprozesses zu bearbeiten und zu realisieren. Viele dieser **Ziele der Gestaltung des Arbeitsprozesses** und die erwünschten Handlungsmöglichkeiten müssen bereits in der Software angelegt werden.

- Die Software muß für das Erreichen der definierten Ziele geeignet sein. Sie muß die Erstellung der Arbeitsergebnisse in der gewünschten Form und Qualität unterstützen, sie muß andere qualitative Ziele des Arbeitsprozesses wie Flexibilität und Schnelligkeit ermöglichen.
- Die für die Erstellung der Arbeitsergebnisse notwendigen Arbeitsverfahren und -methoden müssen mit der Software durchführbar sein. Diese Forderung bezieht sich keineswegs nur auf den algorithmisierbaren Anteil dieser Verfahren sondern auch auf die unstrukturierten, nicht exakt beschreibbaren Anteile.
- Die Software muß alle in das Arbeitsergebnis eingehenden Vorprodukte und Materialien weiterverarbeiten können.
- Die Software muß für die Zusammenarbeit der Anwender in der Erstellung gemeinsamer Arbeitsergebnisse geeignet sein.

- Die Software muß mit den Fähigkeiten und dem Wissen der Anwender einsetzbar sein.

Andere arbeitsgestaltende Aspekte des Software-Projekts realisieren sich erst in der **Implementierungsphase** selbst. Um hier einen zweckgerichteten und effektiven Einsatz der Software einzuleiten, ist es notwendig, die technische Implementierung mit der individuellen und kooperativen Aneignung der Software durch die Anwender zu verknüpfen. Mit dem Einsatz einer neuen Software haben sich im Arbeitsprozeß die Zwecke der Arbeit und die Bedeutungen bzw. Bedeutungsstrukturen der Arbeitsmittel und -gegenstände sowie die Arbeitsverfahren z.T. erheblich verändert. Sie sind weder allein aus der äußeren Form der Software noch über eine Einführung in ihren Funktionsumfang zu erschließen. Darüber hinaus erfordert der Umgang mit der Software ein umfassendes Wissen über den Arbeitsprozeß, wie er im Software-Projekt projektiert wurde. Auf dieser Basis muß der Arbeitsprozeß in der Implementierungsphase von den Mitarbeitern des Unternehmens neu erarbeitet werden. Das Software-Projekt muß also dem Anwender-Unternehmen die **der Entwicklung der Software zugrundeliegenden Annahmen** zur Verfügung stellen können, um einen effektiven Einsatz der Software sicherzustellen. Benötigt werden

- eine Beschreibung der Tätigkeiten, ihrer Zwecke und der Verrichtungsweisen unter Einbeziehung der Nutzung der Software,
- eine Beschreibung der Software-Objekte und ihrem Bezug zu Tätigkeiten und Kooperationsverhältnissen,
- eine Beschreibung der Kooperationsverhältnisse und ihrer Koordinationsformen, wie sie sich in Bezug auf die Arbeitstätigkeiten und die Softwareobjekte darstellen.

Die **Formulierung dieser Vorstellungen bezeichnen wir als Theorie** des Arbeitsprozesses. In den der Implementierung vorgelagerten Phasen des Projekts wird der Arbeitsprozeß theoretisch, d.h. als eine allgemeine Antizipation von Zwecken, Tätigkeiten und Kooperationsverhältnissen gestaltet. Diese Theorie bildet die systematische Grundlage für die Herausbildung subjektiver Bezugsschemata in der Arbeit und eines verallgemeinerten Bezugsschemas in der Kooperation. Darüber hinaus dient sie als Bezugsschema für die Realisierung der Software, indem Gestaltungsanforderungen an die Software hier systematisch aus den Zwecken des Arbeitsprozesses abgeleitet sind.

3.3.2 Prototyping

Der bekannteste und auch in der Literatur am besten dokumentierte Ansatz prozeßorientierten Vorgehens in der Software-Entwicklung ist das Prototyping, die **Entwicklung und der Einsatz von frühen Versionen des zukünftigen Software-Systems**.

Budde u.a. charakterisieren Prototyping so:

"- Prototyping is an approach based on an evolutionary view of software development and having an impact on the development process as a whole.

- Prototyping involves producing early working versions (prototypes) of the future application system and experimenting with them.

- Prototyping provides a communication basis for discussions among all the groups involved in the development process, especially between users and developers.

- Prototyping enables us to adopt an approach to software construction based on experiment and experience. [Budde u.a. 1992, S. 6]

Mit der Entwicklung und dem Einsatz von Prototypen werden verschiedene **Ziele** verfolgt, die sich im Projekt ergänzen:

Prototypen dienen als Kommunikationsmedium in der Kooperation von Anwendern und Entwicklern. Sie erleichtern die Auseinandersetzung mit dem Arbeitsprozeß und den Gestaltungsanforderungen an die Software, indem sie einen gemeinsamen Bezugspunkt bieten. Dieses Ziel steht im **explorativen Prototyping** im Vordergrund.

Prototypen ermöglichen in sehr frühen Projektphasen, sich technische Gestaltungsmöglichkeiten und ihre Auswirkungen auf die Arbeitsgestaltung erfahrungsorientiert anzueignen. Sie bieten damit für Anwender und Entwickler gleichermaßen die Möglichkeit, Gestaltungsanforderungen schrittweise zu bestimmen, zu realisieren und zu bewerten. Prototyping mit dieser Zielstellung wird als **experimentelles Prototyping** bezeichnet.

Schließlich wird es im Rahmen einer evolutionären Sicht von Entwicklungsprozessen zunehmend als sinnvoll erachtet, Software-Systeme inkrementell zu entwickeln, also in einem stufenweisen Übergang von sehr einfachen Prototypen

bis hin zum fertigen Produkt. Hier wird von **evolutionärem Prototyping** gesprochen. [vgl. Spitta 1989, S. 5][194]

In der Entwicklung und dem Einsatz von Prototypen im Software-Projekt ist es weiterhin sinnvoll, zwischen horizontalem und vertikalem Prototyping zu unterscheiden. Mit dem **horizontalen Prototyping** wird nur eine Ebene der Software, im allgemeinen die Benutzungsschnittstelle ausgearbeitet. Eingabemasken, Menüs oder andere Visualisierungen der geplanten Funktionalität der Software sollen einen Eindruck der Software vermitteln und als Kommunikationsmedium mit den Anwendern dienen.

Mit dem **vertikalen Prototyping** wird ein ausgewählter Teil des Zielsystems entwickelt und den Anwendern in einer lauffähigen Version vorgestellt. [vgl. Budde u.a. 1992, S. 39] Diese Form des Prototyping unterstützt einerseits die Entwickler bei der Einschätzung der technischen Machbarkeit ihrer Ideen, da sie ihre Ideen bis auf die unterste Ebene der Software realisieren müssen. Andererseits ermöglichen diese Prototypen Anwendern echte Erfahrungen in realen oder simulierten Arbeitsprozessen, da sie als lauffähige Versionen - wenn auch nur in eingeschränkter Funktionalität - einsetzbar sind.

In seinen verschiedenen Formen ist Prototyping ein Ansatz, der die Realisierung des Produkts »Software« systematisch mit der Anforderungsdefinition und den Lernprozessen von Anwendern und Entwicklern verknüpft. Über den experimentellen Einsatz von z.B. partiell lauffähigen Versionen können Erfahrungen gesammelt und ausgewertet werden sowie weiterführende Gestaltungsanforderungen konkretisiert oder auch revidiert werden. Werden diese Aspekte des Einsatzes von Prototypen mit der Theoriebildung über den Arbeitsprozeß verknüpft, so können wir auch von einem **erfahrungsorientierten Lernen über den zukünftigen Arbeitsprozeß** sprechen.

4 Zusammenfassung

Im Software-Projekt muß heute die enge Verknüpfung von Arbeitsgestaltung und Software-Entwicklung sowie die Zusammenarbeit von Anwendern und Entwicklern bewältigt werden. Beide Aspekte führen dazu, daß die Anforderungen an eine aufgabenangemessene Steuerung von Software-Projekten zunehmen.

- Software-Konstruktion und Arbeitsgestaltung stehen in einem reziproken Verhältnis und können gemeinsam nur in engen Rückkopplungsschleifen

[194] Dieser Überblick, der in der Literatur sehr häufig zitiert wird, geht auf Christiane Floyd zurück. [Floyd 1984, S. 38]

geleistet werden. Ursache dafür ist die zunehmende Komplexität im Verhältnis der lebendigen Arbeit zu ihren formalisierten und abgebildeten Anteilen.

- Die individuellen Lernprozesse, die Entwickler und Anwender im Projekt durchlaufen müssen, um sich ein Verständnis des Arbeitsprozesses zu erarbeiten und Gestaltungsanforderungen an die Software und an die Arbeit zu definieren, weichen aufgrund der sehr unterschiedlichen Ausgangsbedingungen voneinander ab und müssen im Prozeß aufeinander bezogen werden.

Das traditionelle Software Engineering mit seinen ihm zugrundeliegenden phasen- und produktorientierten Vorgehensmodellen sind diesen Prozessen nicht mehr angemessen. Die grundlegenden Annahmen dieser Ansätze stehen im direkten Gegensatz zur Realität in vielen Software-Projekten. Hier wird unterstellt, daß

- die Gestaltungsanforderungen vorab und unveränderlich gegeben sind,
- das Verhältnis zwischen Software und lebendiger Arbeit klar strukturiert und über wenige Schnittstellen zu regeln ist und
- der Prozeß in einer sequentiellen Abfolge von Phasen, die der Erstellung vordefinierter Produkte dienen, abzubilden und anhand eines solchen Modells zu steuern ist.

Diese Annahmen stellen in modernen Software-Projekten eine Behinderung dar, indem sie den evolutionären Verlauf des Prozesses verdecken und letztlich verhindern. Statt einer produktorientierten Sicht ist heute eine prozeßorientierte Sicht notwendig, die zyklische Bewegungen des Projekts fundiert. Diese prozeßorientierte Sicht spiegelt sich in einigen modernen Ansätzen des Software Engineering wider:

Zyklische Projektmodelle wie das STEPS-Modell von Floyd bieten den notwendigen Raum für Rückkopplungsmöglichkeiten und sind damit die Basis eines prozeßorientierten Vorgehens.

Die Strukturierung des Prozesses über Referenzlinien statt über Meilensteine ermöglicht variables und situationsabhängiges Handeln vor einem kontrollierbaren Planungshorizont.

Nur wenige Produkte werden als feste Bestandteile des Software-Projekts angesehen. Sie werden im Spannungsfeld von Arbeitsgestaltung und Software-Entwicklung definiert und hinsichtlich ihres Entstehens im kooperativen Lernprozeß betrachtet:

- Die Definition von Gestaltungsanforderungen an die Software wird in einen Prozeß der Theoriebildung eingebettet, in dem das komplexe reziproke Verhältnis von Arbeitsgestaltung und Software-Entwicklung zu bewältigen ist.
- Die Software-Realisierung wird in einen Prozeß des Prototyping eingebettet und damit einem erfahrungsorientierten Lernen über den zukünftigen Arbeitsprozeß zugänglich gemacht.

Mit diesen Ansätzen kann ein Methodenrahmen für moderne Software-Projekte abgesteckt werden, der einerseits den Anforderungen an Planbarkeit und Kontrollierbarkeit der Software-Entwicklung genüge tut, andererseits aber auch den Raum für flexibles und kreatives Arbeiten im Software-Projekt bietet, in dem ein neuer Arbeitsprozeß mit neuer Softwareunterstützung entstehen kann.

Teil C
Qualifizierungsbaustein »Aufgabenverständnis«
IT-Fachkräfte im Spannungsfeld zwischen Technikgestaltung und Arbeitsgestaltung

Einleitung[*)]

Die Aufgaben des Qualifizierungsbausteins: Qualifizieren und Orientieren

IT-Fachkräfte werden durch die aktuelle Entwicklung vor immer neue Aufgaben gestellt. Die Anforderungen an diese Berufsgruppe sind einem tiefgreifenden Wandel unterworfen, dem die Beschäftigten in diesem Bereich nur durch den Erwerb umfassender neuer Kompetenzen begegnen können.

Diese These ist inzwischen Allgemeingut. In Diskussionen mit Beschäftigten und Betriebsräten wird sie ebenso vertreten und mit den jeweils nächstliegenden Praxisbeispielen erläutert, wie in Expertengesprächen mit Personalentwicklern oder Qualifizierungsanbietern in der IT-Branche.

Die Schlüsselgruppe »IT-Fachkräfte« erhält in den modernen Innovationsprozessen der Unternehmen, wie sie gegenwärtig unter dem Stichwort »Lean Production« diskutiert werden, eine vollkommen neue Rolle. In dem Maße, wie das Erstellen von Computersystemen immer zugleich Technik- und Arbeitsgestaltung ist und nur noch im Rahmen einer Entwicklungspartnerschaft mit den Menschen in den Anwenderabteilungen machbar ist, müssen die technischen Qualifikationen der Computerspezialisten mit sozialen Qualifikationen zusammenkommen.

Dieser Anforderungswandel erfordert eine Neuorientierung der Aus- und Weiterbildung von Computerfachkräften auf ein neues Qualifizierungsziel. An die Stelle des »Technikers« tritt die »Ganzheitliche Arbeitsgestaltungskompetenz« als Leitbild für die konzeptionelle Ausrichtung der Aus- und Weiterbildung. Hierzu wird die Qualifizierung so verändert, daß sie ein modernes, ganzheitliches Verständnis der Aufgaben von IT-Fachkräften vermittelt. Bezogen darauf werden zentrale Qualifizierungsinhalte wie »Methoden der Systementwicklung« und »Projektmanagement« systematisch vernetzt. Das Aus- und Weiterbildungsangebot sowie die Qualifizierungsnachfrage werden dabei gleichermaßen verändert.

Unterstützt wird dieser Veränderungsprozeß durch die Entwicklung neuer Qualifizierungsbausteine und -systeme, durch Prozesse der Dozentenqualifizierung und eine gezielte Unterstützung von Organisationsentwicklungsprozessen. Die

---

[*)] Dieser Qualifizierungsbaustein basiert auf einer Arbeit, die von Andrea Baukrowitz, Andreas Boes und Christian Boß erstellt wurde.

Prozesse zur Veränderung des Nachfrageverhaltens werden durch Personalentwicklungsinstrumente und -konzepte sowie Qualifizierungsmaßnahmen für Personalentscheider begleitet.

Dem Qualifizierungsbaustein »Aufgabenverständnis« liegt der Anspruch zugrunde, ein modernes, ganzheitliches Verständnis der Aufgaben von IT-Fachkräften inhaltlich zu füllen. Dies heißt zunächst, inhaltlich zu bestimmen, welche zentralen Aufgaben sich für IT-Fachkräfte aus der Forderung ergeben, Informationssysteme zu entwickeln, deren Qualität den aktuellen Anforderungen der Unternehmen entspricht.

Flexibilität und hohe Reaktionsgeschwindigkeit gegenüber sich rasch wandelnden Markterfordernissen müssen sich mit der Fähigkeit verbinden, in einer Organisation mit verteilter Verantwortung und Entscheidungskompetenz an allen Arbeitsplätzen die entscheidungsrelevanten Informationen zeitnah zur Verfügung zu stellen.

Daraufhin gilt es zu entwickeln, welche Anforderungen mit diesen Entwicklungsaufgaben an IT-Fachkräfte gestellt und welche Kompetenzen benötigt werden, um diesen Anforderungen genügen zu können. Den Hintergrund dieser Untersuchung liefert eine Analyse der historischen Entwicklung der Informationstechnik unter besonderer Berücksichtigung des Spannungsverhältnisses zwischen Arbeit und Technik bei der Entwicklung von IuK-Systemen.

Technikgestaltung im Kontext der Arbeitsgestaltung und die Kooperation verschiedener Expertengruppen in interessengeleiteten sozialen Prozessen sind die Kernanforderungen, denen IT-Fachkräfte gegenüberstehen. Die Notwendigkeit, in dieser Perspektive das Dreieck von Technikgestaltungskompetenz, Kooperationskompetenz und Prozeßkompetenz in einem ganzheitlichen Konzept an IT-Fachkräfte zu vermitteln, beschreibt den allgemeinen Fluchtpunkt für die Gestaltung der Qualifizierungsprozesse für diese Gruppe.

Beim Erwerb dieser Kompetenzen und ihrer Umsetzung in der gegenwärtigen oder künftigen beruflichen Arbeit orientiert dieser Qualifizierungsbaustein »Aufgabenverständnis« Auszubildende und Weiterbildungsteilnehmer im Berufsfeld Informatik über die allgemeinen Zusammenhänge zwischen Anforderungen und notwendigen Kompetenzen.

Soweit diese allgemeinen Zusammenhänge in der Qualifizierungspraxis in konkrete Handlungssituationen eingebunden und mit konkreten Handlungsorientierungen im Ausgang dieses Bausteins verbunden werden, wirkt die Vermittlung des Fluchtpunkts »Ganzheitliches Aufgabenverständnis« über diesen Qualifizierungsbaustein handlungsleitend für seine Zielgruppen:

- Dozenten in der Aus- und Weiterbildung von IT-Fachkräften,
- Entwickler von Qualifizierungsprogrammen, Curricula und Modulen für die Aus- und Weiterbildung von IT-Fachkräften,
- Entscheider im Bereich der Qualifizierungs- und Personalentwicklungsplanung und -realisierung für IT-Fachkräfte,
- Teilnehmer von Aus- und Weiterbildungsmaßnahmen für IT-Fachkräfte.

Um diese Aufgaben für die Vielfalt der Adressaten erfüllen zu können, muß der Baustein zwei Anforderungen genügen:

Es handelt sich um einen **allgemeinen** Qualifizierungsbaustein, der unabhängig von der Zielgruppe, für den er eingesetzt werden soll, grundlegend den Wandel der Aufgaben von IT-Fachkräften vermittelt.

Es handelt sich um einen **konkreten** Qualifizierungsbaustein, der von einer konkreten Handlungssituation der jeweiligen Teilnehmer ausgeht und in einer konkreten Handlungsorientierung für die Teilnehmer in ihrer Situation mündet.

Der Aufbau des Qualifizierungsbausteins:
Das Verhältnis von Allgemeinem und Besonderem

Der »Qualifizierungsbaustein Aufgabenverständnis« ist in insgesamt fünf Arbeitseinheiten (AE) gegliedert. Diese Arbeitseinheiten sind in sich wiederum in Arbeitsschritte (AS) unterteilt, die die jeweils notwendigen Erkenntnisschritte in ihren Zielen, Inhalten und Methoden beschreiben. Je nach dem Umfang der zu erarbeitenden Inhalte sind die Arbeitsschritte bei Bedarf weiter in einzelne Lernschritte geteilt. Diese Lernschritte sind in der Gliederung nicht gesondert ausgewiesen.

Die Arbeitseinheiten

AE 2 Die Entwicklung der Informationstechnik

AE 3 Aufgaben von IT-Fachkräften

AE 4 Aufgabenverständnis und Kompetenzen von IT-Fachkräften

bilden dabei den **allgemeinen Teil** des Qualifizierungsbausteins.

Dieser allgemeine Teil vermittelt ein ganzheitliches Aufgabenverständnis für IT-Fachkräfte, das für die Orientierung im Beruf und die Ausrichtung individueller und gruppenbezogener Qualifizierungsstrategien eine angemessene Handlungsorientierung liefert.

Die Arbeitseinheiten

AE 1 Einstiegsbaustein »Software Engineering« und

AE 5 Ausgang »Methoden des Software Engineering«

sind dagegen an der beruflichen- oder Ausbildungssituation der Teilnehmer einer Unterrichtseinheit oder eines Lehrgangs orientiert, sie bilden den **besonderen Teil** des Qualifizierungsbausteins.

Die AE 1 entwickelt mit den Teilnehmern die situationsbezogene Lernmotivation zur Auseinandersetzung mit den allgemeinen Grundlagen des Aufgabenverständnisses.

Die AE 5 bietet den Teilnehmern eine handlungsleitende Orientierung für ihre konkrete Situation und ermöglicht ihnen in der Lernsituation die Auseinandersetzung mit möglichen praktischen Konsequenzen des erarbeiteten Wissens.

AE 1 und AE 5 müssen je nach der Einsatzumgebung des Qualifizierungsbausteins variiert und an die jeweilige Zielgruppe angepaßt werden.

AE 1 Einstiegsbaustein »Software Engineering«

Skizze eines Einstiegs für eine Unterrichtseinheit/einen Lehrgang, der eine Handlungsorientierung in Richtung **»Erwerb einer angemessenen Qualifizierung und Handlungskompetenz im Bereich Methoden der Softwareentwicklung«** zum Ziel hat.

Vorbemerkung

Im Einstiegsmodul sollen die Teilnehmer motiviert werden, sich über den gesamten Qualifizierungsbaustein mit den Trends in der Veränderung der Systementwicklung, den Trends in den Veränderungen der Anforderungen an IT-Fachkräfte und den entsprechenden beruflichen und/oder qualifikatorischen Konsequenzen im Handlungsfeld »Methoden der Softwareentwicklung« auseinanderzusetzen.

Im Einstiegsmodul sollen möglichst gemeinsam mit den Teilnehmern zielführende Fragen formuliert werden, die im weiteren Verlauf des Qualifizierungsmoduls Schritt für Schritt bearbeitet und in der abschließenden Arbeitseinheit möglichst konkret und handlungsleitend beantwortet werden.

AS 1 Die aktuelle Situation in der Softwareentwicklung

Ziel:

Herausarbeiten der aktuellen Situation in der Softwareentwicklung.

Inhalt:

Auf Grundlage des Artikels über den Beitrag Tom DeMarcos zur Situation in der Softwareentwicklung wird die aktuelle Krise in der Softwareentwicklung herausgearbeitet. [Tom DeMarco in VDI Nachrichten 16/1990, S. 34]

Anhand der These "Kein einziges Software-Projekt ist an technischen Schwierigkeiten gescheitert. (...) Die meisten Probleme sind nicht technischer, sondern sozialer Natur" von Tom DeMarco und anhand der These "Software-Design ist immer auch Sozial-Design" von Joseph Weizenbaum werden die Ursachen und Hintergründe dieser Entwicklung problematisiert. [Weizenbaum-Interview in ComputerInformation 7-8/1992, S. 3-6]

AS 2 Anforderungen an die Entwickler und die Entwicklungsmethoden

Ziel:

Einschätzungen der Teilnehmer zu den Anforderungen an die Softwareentwickler und die Entwicklungsmethoden formulieren.

Inhalt:

Auf der Grundlage der Artikel aus dem AS 1 formulieren die Teilnehmer Einschätzungen zu den Fragen:

Welche neuen Anforderungen kommen in dieser Situation auf Softwareentwickler zu?

Wie müssen angemessene Methoden der Softwareentwicklung ausgelegt sein?

Aus den formulierten uneinheitlichen und teilweise spekulativen Einschätzungen wird dann die Notwendigkeit begründet, sich weiter mit der Entwicklung der Anforderungen an Softwareentwickler und Methoden der Softwareentwicklung auseinanderzusetzen.

Unterstützend können an dieser Stelle die Argumente aus der Auseinandersetzung von Wolfgang Coy mit der Entwicklung der Informatik als Wissenschaftsdisziplin [Coy 1992b] von den Dozenten in die Diskussion eingebracht werden.

AS 3 Leitfragen und Aufbau des Qualifizierungsbausteins »Aufgabenverständnis«

Ziel:

Erarbeiten der Leitfragen für das Qualifizierungsmodul und Erläuterung und Begründung des weiteren Vorgehens.

Inhalt:

In diesem Arbeitsschritt wird herausgearbeitet, daß die vorhandenen Probleme sinnvoll zu bearbeiten sind, wenn man in der Lage ist, folgende Fragen zu beantworten:

> Wohin entwickeln sich die Trends in der Systementwicklung und welche Faktoren bestimmen diese Entwicklung?
>
> Wie entwickeln sich die Aufgaben von IT-Fachkräften und was bestimmt diese Entwicklung?

Was bedeuten diese Entwicklungen für die notwendigen Kompetenzen von IT-Fachkräften?

Aus diesem Material kann man dann Antworten auf folgende Fragen finden:

Welche Konsequenzen lassen sich daraus für das Handlungsfeld »Methoden der Softwareentwicklung« formulieren?

Im Übergang zum Qualifizierungsbaustein wird kurz der weitere Aufbau der Unterrichtseinheit vorgestellt und erläutert:

- Entwicklung von Informationstechnik und Anforderungen an Computerfachkräfte bis heute (AE 2).
- Die aktuellen Trends und ihre Wirkung auf die Aufgaben von IT-Fachkräften (AE 3).
- Die Entwicklung von Kompetenzen im Spannungsfeld von Arbeit und Technik (AE 4).
- Konsequenzen für das Handlungsfeld »Methoden der Softwareentwicklung«.

AE 2 Die Entwicklung der Informationstechnik

Vorbemerkung

Dieser Teil soll den Blick für die wesentlichen aktuellen Veränderungen in den Systementwicklungsprozessen öffnen und die Motivation für eine vertiefte Auseinandersetzung mit den künftigen Aufgaben und der künftigen Rolle von Computerfachkräften schaffen. Dazu bedarf es der Erarbeitung der historischen Entwicklung der Informationstechnik, um aus einem Vergleich der verschiedenen Entwicklungsabschnitte das grundlegend Neue im Verhältnis von Arbeitsgestaltung und Technikgestaltung sowie die grundlegend neuen Anforderungen an Computerfachkräfte begründet herausarbeiten zu können.

AE 2.1 Darstellung und Begründung der Ziele und Inhalte der AE 2

AS 1 Einstieg ins Thema: Ein Vergleich zwischen »Damals« und »Heute«

Ziel:

Die Teilnehmer sollen zur Auseinandersetzung mit historischen Aspekten der Entwicklung der Anforderungen an Computerfachkräfte motiviert werden.

Inhalt:

Zunächst wird den Teilnehmern anhand der technischen Entwicklung demonstriert, wie sich eine vielfältige Struktur technischer Elemente in den Betrieben verbreitet und dort an nahezu sämtlichen Arbeitsplätzen ihren Einzug gehalten hat.

Dann wird den Teilnehmern durch die Gegenüberstellung des alten Labortechnikers mit einer »modernen« DV-Fachkraft, die mit einer Gruppe von Benutzern engagiert Probleme ihrer Arbeit diskutiert, die gewaltige Entwicklung und der große Unterschied zwischen »Damals« und »Heute« in der Arbeit von IT-Fachkräften demonstriert.

AS 2 Das Thema: IT-Fachkräfte im Spannungsfeld zwischen Arbeit und Technik

Ziel:

Die Teilnehmer sollen die Arbeitsmethode bei der Untersuchung der historischen Entwicklung kennenlernen.

Inhalt:

Zu der Arbeit an der historischen Einordnung der Entwicklung wird folgendermaßen übergeleitet:

- Die technische Entwicklung kann man relativ einfach beschreiben und in Zahlen und Meßwerten vor allem bei der Hardware ausdrücken. Dies ist vielleicht auch schon im Unterricht passiert, vielleicht sogar schon auf der Schule.
- Selten oder nie wird die Situation bearbeitet, in der die Computerfachkräfte mit ihrer Arbeit in den Phasen dieser Entwicklung stehen, was ihre Anforderungen zu Beginn der Entwicklung ausmacht und wie sich diese Anforderungen in einem dynamischen Prozeß bis heute entwickelt haben.
- Was erst recht nicht in den Blick rückt und ausführlich bearbeitet wird, ist das Verhältnis zwischen der Arbeit, die mit Hilfe des Rechners erledigt wird, der Technik, die diese Aufgabe leistet bzw. leisten soll - und häufig nicht leistet - und den Computerfachkräften, deren Anforderungen ja immer ersteinmal darin bestehen, dieses Verhältnis irgendwie in den Griff zu bekommen (»Abbildungsproblematik«).
- Genau die Entwicklung dieses Dreiecks Arbeit, Technik, Computerfachkräfte werden wir in diesem ersten Schritt bearbeiten, um zu sehen, wie sich die Trends in diesem Verhältnis entwickelt haben, was diese Trends bestimmt und die Entwicklung in Schwung hält. Im Anschluß daran wird eine Untersuchung möglich, wie sich begründete Vorstellungen darüber entwickeln lassen, was (mindestens in den nächsten zehn Jahren) im Berufsfeld Informatik an Entwicklung angelegt ist und wie man sich darauf in der allgemeinen Richtung einstellen kann.

Wenn die Teilnehmer aus ihrer eher technikfixierten Denkweise hier Widerstände gegen diese Betrachtung und diesen Zugang entwickeln (Tenor: »Aber es ist doch die Technik und die Software, die diese Entwicklung bestimmen!«), wird den Teilnehmern an einem historischen Beispiel verdeutlicht, daß es beim Einsatz von Technik primär um die Effektivierung von menschlicher Arbeit geht und wie Computerfachkräfte mit dieser Abbildungsproblematik befaßt sind.

AE 2.2 Der Prozeß der Entwicklung der Informationstechnik

Vorbemerkung:

Die Abbildungsproblematik läßt sich den Teilnehmern aus ihrer Ausbildungserfahrung mit der Entwicklung von Software erschließen. Daneben verfügen sie wahrscheinlich über abrufbare Kenntnisse über den Technikkontext der Entwicklung.

Die Schwerpunkte des inhaltlichen Inputs in dieser Arbeitseinheit liegen daher in der Einführung und der (historischen) Entwicklung des Nutzungskontextes und des Aufgabenkontextes.

AS 1 Das Instrument zur Untersuchung der Entwicklung

Ziel:

Erarbeitung der Struktur von Abbildungsproblematik, Nutzungs-, Technik- und Aufgabenkontext als Analyseinstrument für die Entwicklung der Informationstechnik im Betrieb.

Inhalt:

Wenn dies nicht schon am Ende des ersten Teils gemacht wurde, wird durch die Dozenten an einem historischen Beispiel erläutert, daß die Problematik der Abbildung von Arbeit auf Technik ständig die Arbeit von Computerfachkräften durchzieht. Daran anschließend wird diese Situation an Ausbildungs- und/oder Praktikumssituationen weiter vertieft.

Aus diesen Beispielen heraus werden die Kontext-Kategorien und ihre grundsätzlichen Zusammenhänge untereinander entwickelt:

- Der Nutzungskontext
  beschreibt das jeweilige Verhältnis von Arbeit und Technik, das im Ergebnis der Abbildung von Arbeit oder Arbeitsstrukturen auf Technik entsteht.
- Der Technikkontext
  beschreibt den jeweiligen Stand der Hardware- und Softwaretechnik, mit dem die Abbildungsproblematik gelöst werden kann (oder auch nicht).
- Der Aufgabenkontext
  beschreibt das Verhältnis der Aufgabe »Arbeitsgestaltung« zur Aufgabe »Technikgestaltung« im Aufgabenspektrum der Computerfachkräfte.

AS 2 Die Ausgangssituation: Der isolierte Großrechner

Ziel:

Erarbeiten der historischen Ausgangssituation der Entwicklung der Informationstechnik.

Inhalt:

Anhand von Fallbeispielen wird zunächst das Problem der Abbildung von Arbeit auf den Computer bearbeitet. Ausgehend von der Abbildungsproblematik

- »Algorithmisieren« von (Teilen) von Arbeitsstrukturen und Abbildung in Programmen

wird der Nutzungskontext

- Trennen/Herauslösen von Arbeit und Übertragung auf den Rechner

und der Technikkontext

- Großrechenanlagen isoliert in Rechenzentren ohne direkte Verbindung zu den Arbeitsplätzen

erarbeitet.

Anschließend wird die Frage bearbeitet:

Welche Aufgaben übernehmen in diesem Entwicklungszusammenhang die Computerfachkräfte?

- Bei der Gestaltung der Arbeit entwickeln sie höchstens Anforderungen/Hinweise zur rechnergerechten Gestaltung von Restarbeit,
- bei der Gestaltung der Technik erledigen sie das Algorithmisieren und Programmieren von starren Arbeitsstrukturen mit dem Ziel: »Optimiere ein technisches System«.

AS 3 Die Etappe der Durchsetzung der Dialogverarbeitung

Ziel:

Das Verhältnis von Arbeit und Technik als zentrales Moment der historischen Entwicklung erarbeiten.

Erarbeiten des Veränderungsprozesses in den Anforderungen an die IT-Fachkräfte.

Inhalt:

Anhand eines Fallbeispiels zur Durchsetzung der Dialogverarbeitung als typischer Form der Datenverarbeitung werden die Fragen bearbeitet:

Wie verändert sich die Abbildungsproblematik?

- Es werden zunehmend fall- oder vorgangsbezogene Entscheidungsstrukturen auf den Rechnern abgebildet.

Wie verändert sich der Nutzungskontext?

- Arbeit wird in Arbeitsteilung zwischen Mensch und Computer zunehmend im programmbestimmten Dialog Mensch - Rechner erledigt. Die Arbeiten, die an den Arbeitsplätzen im Dialog mit dem Rechner erledigt werden müssen, werden durch das Programm bestimmt und strukturiert.

Wie verändert sich der Technikkontext?

- Die Großrechner erhalten über Terminals eine Verbindung zu einer ständig steigenden Zahl von Arbeitsplätzen.

Wie verändern sich mit dieser Entwicklung die Aufgaben der Computerfachkräfte?

- Sie haben die optimale Verbindung und Anpassung zwischen Großrechner und Arbeitsplätzen herzustellen. Die Gestaltung der entsprechenden Systeme kann nun nicht mehr als Laboraufgabe an reine Techniker delegiert werden. Die in der Logik der Dialoganwendungen angelegte »Abbildung betrieblicher Strukturen« erfordert zum einen Kenntnisse des Anwendungsfeldes (Stichwort Fachinformatiker), zum anderen setzt die adäquate Abbildung komplexer betrieblicher Strukturen zunehmend den Einsatz ingenieursmäßiger Methoden der Systementwicklung voraus.
- Anforderungen der Benutzer spielen in zunehmendem Maße eine Rolle. Vor allem bei komplexeren Aufgaben ist eine Beteiligung der Benutzer Voraussetzung für eine Dialoggestaltung durch die Computerfachkräfte (Stichwort »Systemanalyse«). Die Aufgaben der Arbeitsgestaltung nehmen zu

(Stichworte: »Schnittstelle Mensch-Maschine«, »Ergonomie«). Das Gestaltungsziel bleibt weiterhin: »Optimiere ein technisches Dialogsystem«.

AS 4 Arbeit und Technik als »Roter Faden« der Entwicklung

Ziel:

Zusammenfassend herausarbeiten, daß Arbeit und Technik und deren gegenseitige Durchdringung das wesentliche Moment ist, das die Anforderungen an IT-Fachkräfte verändert.

Inhalt:

Aus den Ergebnissen der beiden vorherigen Schritte wird herausgearbeitet:

Es handelt sich um einen historischen Prozeß der zunehmenden Verbindung und Durchdringung von Arbeit und Technik.

Diese Veränderung des Verhältnisses von Arbeit und Technik bestimmt zunehmend die Veränderung der Aufgaben von IT-Fachkräften. Aufgaben aus dem Bereich Arbeitsgestaltung gewinnen an Bedeutung.

In der nächsten Arbeitseinheit wird dieser rote Faden »Arbeit und Technik« weiter bearbeitet, um die aktuellen Trends der Veränderungen von Anforderungen herauszufinden.

AE 3 Aufgaben von IT-Fachkräften

Vorbemerkung

Im Qualifizierungsbaustein »Aufgabenverständnis« dient die Arbeitseinheit »Aufgaben von IT-Fachkräften« der gemeinsamen Erarbeitung eines Bildes von Aufgaben und von der Arbeitsrealität, welches das Verhältnis von Arbeit und Technik zum Ausgangspunkt hat.

Die Arbeitseinheit baut auf folgendem Wissensstand auf:

- Die Informationstechnik und die Arbeit der IT-Fachkräfte ändert sich.
- Auf der Grundlage einer groben Einschätzung der Richtung der Veränderung ist als Kern der Entwicklung das Verhältnis Arbeit und Technik identifiziert.

Bei den Teilnehmern kann man zur Erarbeitung der inhaltlichen Zusammenhänge in dieser Arbeitseinheit folgende Motivation annehmen:

- Sie wollen die aktuelle Situation besser einschätzen können, um sich selbst und die eigene berufliche Orientierung darin zu verorten.
- Sie suchen allgemein nach einer Handlungsorientierung.
- Sie wollen wissen, wie sie sich dafür qualifizieren müssen.

AE 3.1 Darstellung und Begründung der Ziele und Inhalte der AE 3

AS 1 Zusammenfassung des Standes der Arbeit

Ziel:

Durch die Zusammenfassung der vorangegangenen Arbeitseinheit transparent machen, auf welchem Wissensstand die folgende AE aufbaut.

Inhalt:

- Die Informationstechnik und die Arbeit der IT-Fachkräfte ändert sich.
- Diese Entwicklung ist Resultat des sich verändernden Verhältnisses Arbeit - Informationstechnik.

- Der Umbruch von der Batch-Verarbeitung zum Dialogbetrieb stellte bereits eine grundlegende Veränderung des Verhältnisses Arbeit - Informationstechnik dar.
- Heute erleben wir wieder grundlegende Veränderungen.
- Die Teilnehmer sind auf dem Arbeitsmarkt und in den Unternehmen mit diesen Veränderungen konfrontiert.
- Das Raster Nutzungskontext, Technikkontext, Arbeitskontext und Aufgabenkontext hilft bei der Analyse im Moment nicht weiter, man kann es mit dem bisherigen Wissensstand nicht eindeutig füllen.

Daher müssen die modernen Aufgaben von IT-Fachkräften näher untersucht werden, damit sich die Teilnehmer darauf besser einstellen können.

AS 2 Überleitung zur Analyse der Aufgaben von IT-Fachkräften

Ziel:

Die Teilnehmer zur weiteren Auseinandersetzung mit den Aufgaben von IT-Fachkräften anregen und damit für die weitere Arbeit motivieren.

Inhalt:

Das Raster aus dem vorangegangenen Teil (Nutzungskontext, Technikkontext, Arbeitskontext und Aufgabenkontext) wird aufgegriffen und zu folgenden Fragen verdichtet:

- Wie verändert sich heute das Verhältnis von Arbeit und Technik?
- Welche Auswirkungen hat das für die Arbeit von IT-Fachkräften?

AE 3.2 Das Verhältnis von Arbeit und Technik

AS 1 Vorstellung eines modernen IT-Konzepts

Ziel:

Die Teilnehmer lernen das IT-Konzept in einem Unternehmen kennen und können die Komplexität seiner Struktur bewerten.

Inhalt:

Um für sich selbst eine konkrete Handlungsorientierung entwickeln zu können, müssen die Teilnehmer das Verhältnis von Arbeit und Technik konkret bestim-

men können. Dazu benötigen sie ein Grundwissen über den aktuellen Stand informationstechnischer Infrastruktur und moderner IT-Konzepte in Unternehmen.

Durch den Dozenten wird die grobe Linie vorgezeichnet: Das Verhältnis von Arbeit und Technik zeichnet sich im Moment durch grundlegende Veränderungen aus. Die Tendenz geht dahin, alle Tätigkeiten in einem Unternehmen durch Computer zu unterstützen und nicht wie bisher nur einzelne Tätigkeiten durch den Computer zu ersetzen.

Anhand eines Fallbeispiels und der Skizze einer informationstechnischen Infrastruktur wird ein modernes IT-Konzept vorgestellt. Dieses Beispiel soll spontan plausibel sein, also weitgehend auf den Teilnehmervoraussetzungen basieren. Diskussionen über technische Spitzfindigkeiten sollen vermieden werden.

Diese Voraussetzungen lassen sich erfüllen, indem die technische Entwicklung in der Darstellung an eine konkrete Unternehmensentwicklung angebunden wird:

- Das Unternehmen hat einen Großrechner.
- Das Unternehmen hatte vorher bereits einige PC-Netze als Insellösungen und viele Stand-Alone-PCs.
- Im nächsten Entwicklungsschritt hat sich das Unternehmen zu einer unternehmensweiten Vernetzung auf PC-Basis unter Einbindung des Großrechners entschlossen.
- Viele verschiedene Systeme, dezentrale, verteilte Ressourcen und der modulare Aufbau aus einer Vielzahl von Einzelteilen machen diese Systeme sehr komplex.
- Diese Situation findet man heute in vielen Unternehmen.

AS 2 Das Verhältnis Arbeit und Technik

Ziel:

Die Teilnehmer sollen Arbeit und Technik als Einheit begreifen.

Inhalt:

a) Um die Informationstechnik systematisch in den Arbeitskontext zu stellen, wird ein technische Szenario dargestellt und Schritt für Schritt um die Menschen erweitert, die mit der Informationstechnik arbeiten.

Der einzelne Mensch sitzt an seinem Schreibtisch und erledigt mit Hilfe des Computers seine Aufgaben. Er nutzt den Computer als umfassendes Werkzeug, das ihn möglichst bei allen Tätigkeiten unterstützt. Jede Veränderung des

Werkzeugs oder gar die Neueinführung des Werkzeugs Computer verändert die Arbeit und die Arbeitsweise des Menschen.

Das Werkzeug muß an die Arbeitsweise des Menschen angepaßt sein, denn nur er kann bestimmen, wie er am besten seine Arbeit verrichten kann. Andererseits muß sich der Mensch an die Möglichkeiten des Computers anpassen, um ihn in seinen Möglichkeiten optimal ausnutzen zu können.

Der einzelne Mensch arbeitet nicht isoliert, sondern in einem Unternehmen mit vielen anderen Menschen zusammen. Er benutzt gemeinsam mit anderen Datenbestände, Ressourcen und Dienstleistungen, er kommuniziert mit anderen Mitarbeitern. In diesem Zusammenhang nutzt er den Computer als Infrastruktur, die das gesamte Unternehmen wie ein Nervensystem durchzieht.

Die Neueinführung oder die Veränderung dieser Infrastruktur hat tiefgreifende Auswirkungen auf die Zusammenarbeit der Menschen. Sie müssen sich in diesem System neu zurechtfinden, sie müssen alle vorhandenen Möglichkeiten nutzen können. Die Kommunikationsstruktur verändert sich. Die Infrastruktur muß an die Anforderungen an Zusammenarbeit, an die gemeinsame Nutzung von Ressourcen und Dienstleistungen und an Kommunikationsmöglichkeiten angepaßt sein. Andererseits müssen sich alle Mitarbeiter auf diese neue Infrastruktur einstellen.

Die Zusammenarbeit der Menschen wird nicht nur durch die Notwendigkeiten der Arbeit sondern auch in hohem Maße durch betriebliche Strukturen geregelt und beeinflußt. Diese Strukturen spiegeln sich in IT-Systemen wider, andererseits beeinflussen IT-Systeme diese Strukturen.

Abteilungsstrukturen, Verfügungsrechte über Daten und Ressourcen, Formen der Arbeitsteilung wirken sich auf IT-Systeme aus. Erweiterte technische Möglichkeiten, veränderte Anforderungen an die optimale Nutzung der IT-Systeme stellen andererseits diese Strukturen in Frage.

b) Problematisierung des traditionell technischen Qualitätsbewußtseins von IT-Fachkräften mit dem Ziel, die Qualität der Arbeit von IT-Fachkräften und von IT-Systemen in einem optimalen Verhältnis von Arbeit und Technik zu sehen.

IT-Fachkräfte verfügen in der Regel über ein sehr hohes Qualitätsbewußtsein, das sich allerdings auf technische Kriterien beschränkt: Laufzeitverhalten, Qualität der Schnittstellen, die Lösung von Schnittstellenproblemen, vielleicht noch Datenschutz- und Datensicherheitsaspekte stehen bei der Frage nach der Qualität im Vordergrund.

Ziel der Diskussion ist es, dieses Qualitätsverständnis zu thematisieren. An Qualitätskriterien wird der Zielerreichungsgrad der eigenen Arbeit gemessen, an ihnen wird die Arbeit ausgerichtet. Das Qualitätsverständnis ist der Kern des Aufgabenverständnisses. Es muß daher zum Ausgangspunkt der Neudefinition von Aufgaben gemacht werden. Hier kann z.B. die Frage, warum auch technisch optimale Systeme in den Unternehmen oft als schlecht oder sinnlos eingestuft werden, das hergebrachte Bild aufbrechen und die Teilnehmer für die Suche nach anderen Qualitätskriterien zugänglich machen.

Das inhaltliche Ziel und der angestrebte Erkenntnisfortschritt besteht darin, die Qualität der Arbeit von IT-Fachkräften und von IT-Systemen in einem optimalen Verhältnis von Arbeit und Technik zu sehen.

c) Das Verhältnis von Arbeit und Technik, die effektive Nutzung und Einsetzbarkeit der Informationstechnik bestimmt die Qualität, die über Erfolg und Mißerfolg der Arbeit von IT-Fachkräften entscheidet.

In einem kurzen Auswertungsvortrag sollte an dem Gedanken der Komplexität moderner Systeme und der sehr weitgehenden Durchdringung aller Tätigkeiten in einem Unternehmen durch die IT angesetzt werden. Vor diesem Hintergrund werden die in der Diskussion genannten Qualitätskriterien eingeordnet, und zwar beginnend mit den rein technischen Kriterien bis hin zu den arbeitsorientierten Kriterien.

In dieser Auswertung kommt es vor allem auf folgende Inhalte an:

Die Realisierung der technischen Qualität der IT-Systeme ist zwar ein wichtiger Beitrag der IT-Fachkräfte für hochwertige IT-Systeme und nimmt heute noch einen bedeutenden Teil der Arbeit der IT-Fachkräfte in Anspruch. Die Ursache ist v.a. darin zu sehen, daß diese Technologie noch in den Kinderschuhen steckt und demzufolge noch mit vielen Kinderkrankheiten behaftet ist.

Die entscheidende Qualität von IT-Systemen bestimmt sich jedoch aus dem Nutzungskontext. Das Verhältnis von Arbeit und Technik, die effektive Nutzung und Einsetzbarkeit der Informationstechnik bestimmt die Qualität, die über Erfolg und Mißerfolg der Arbeit von IT-Fachkräften entscheidet. Um diese Qualität realisieren zu können, muß die Informationstechnik in ihrem Arbeitskontext Gestaltungsgegenstand von IT-Fachkräften sein.

d) Bereits vorhandenes Wissen und Erfahrungen der Teilnehmer über Kriterien des DV-Einsatzes auf diese Struktur von Qualitätskriterien beziehen.

Das Sammeln von Beispielen für die zwei Ebenen der Qualitätskriterien von IT-Systemen und deren Diskussion soll die Gelegenheit bieten, vorhandenes Wis-

sen und gesammelte Erfahrungen der Teilnehmer auf diese Struktur zu beziehen, und so der Erarbeitung des Verhältnisses von Arbeit und Technik dienen. Dabei soll diese Diskussion immer wieder darauf zugespitzt werden, in welchem Arbeitskontext die Informationstechnik steht.

AE 3.3 Die Verknüpfung von Arbeitsgestaltung und Technikgestaltung

AS 1 Softwarequalität als Schlüsselanforderung an IT-Fachkräfte

Ziel:

Herausarbeiten, daß sich der Anforderungswandel an IT-Fachkräfte über die veränderten Anforderungen und Bedingungen zur Erzeugung von Software-Qualität erschließt.

Inhalt:

Ausgehend von den Inhalten der AE 3.2 wird der zentrale Stellenwert der Qualität der Informationssysteme herausgearbeitet und begründet:

- Im Entwurf und in der Gestaltung komplexer Informationssysteme müssen die Elemente der Arbeitsgestaltung und Technikgestaltung verknüpft werden.
- Die Qualität der Informationssysteme und damit die Qualität der Arbeit von IT-Fachkräften erweist sich erst im Nutzungskontext dieser Systeme.
- Erfolg in der beruflichen Arbeit hat damit grundsätzlich zur Bedingung, diese Qualität bei der konkreten Entwicklung von Systemen herstellen zu können.
- IT-Fachkräfte müssen also im Spannungsfeld zwischen Arbeitsgestaltung und Technikgestaltung zielgerichtet und effizient handeln, um diesen Qualitätsstandard zu erreichen.

  Damit spitzen sich die Fragen nach den aktuellen Veränderungen auf den Qualitätsaspekt zu:

  - Was ist Qualität?
  - Wie sieht das Spannungsverhältnis von Arbeit und Technik konkret aus?
  - Wie erzeuge ich unter diesen Bedingungen Qualität?

AS 2 Die Ebene der individuellen Arbeitsgestaltung

Erster Lernschritt

Ziel:

Ziel dieses Lernschrittes ist die Einführung und Begründung des Leitbildes »Werkzeug« in der Softwareentwicklung.

Inhalt:

Software muß sich heute am Leitbild des Werkzeugs orientieren.

Der Softwareeinsatz wird in den sich verändernden Nutzungskontext in den Unternehmen eingeordnet. Auf dieser Basis der sich wandelnden Anforderungen wird Software als Werkzeug in Abgrenzung zur klassischen Dialoganwendung entwickelt.

1. Welche Aufgaben müssen heute mit Software bewältigt werden?

Von den Unternehmen werden heute vom Markt Flexibilität, Termintreue, kurze Lieferzeiten etc. gefordert. Die konkreten Anforderungen, denen sie begegnen müssen, um wettbewerbsfähig zu bleiben, wechseln rasch (z.B. in Form der Kundenwünsche). In Folge dieser veränderten Anforderungen nimmt an den meisten Arbeitsplätzen der Anteil unstrukturierter, sich ständig ändernder Aufgaben zu.

Von allen Mitarbeitern ist ein hohes Maß an Flexibilität gefordert. Sie müssen mehr Entscheidungen treffen, sie müssen schnell auf Anfragen und Anforderungen reagieren können. Verbunden mit diesen neuen Aufgaben ist ein zunehmender Informationsbedarf. Alle notwendigen Informationen müssen schnell verfügbar sein und verarbeitet werden können.

Unstrukturierte Aufgaben, die sich durch komplexe Entscheidungssituationen und wechselnden Informationsbedarf auszeichnen, entziehen sich weitgehend der Abbildung. Software kann diese Arbeit nur noch unterstützen, nicht mehr ersetzen. Software muß heute ein Werkzeug zur effektiven Informationsbeschaffung und -verarbeitung sein.

2. Software als Werkzeug

Ein Werkzeug ist z.B. ein Hammer, ein Schraubenzieher, ein Meißel. Ein Werkzeug ist die Verlängerung meiner Hand bzw. meiner Sinne und unterstützt mich bei meiner Arbeit. Im Gegensatz dazu steht die Maschine, die unabhängig von menschlicher Einwirkung Tätigkeiten verrichtet (z.B. eine Waschmaschine). Sie ersetzt meine Hand, sie ersetzt einen Teil meiner Arbeit.

Die klassische Dialoganwendung entspricht einer Maschine, die einen Teil der Arbeit - z.B. eines Sachbearbeiters in einem Versicherungsbüro - ersetzt. Dies funktioniert nur bei stark strukturierten Routinearbeiten. Soll der Sachbearbeiter an seinem Arbeitsplatz umfassend unterstützt werden, so benötigt er Software-Werkzeuge, die weniger die Tätigkeiten ersetzen als vielmehr eine Verlängerung des Armes in die Welt der in Form von Bits und Bytes abgelegten Informationen darstellen. Er nutzt die IT, um sich der bereitgestellten Ressourcen für seine Arbeit zu bedienen und um die angebotenen Dienstleistungen etwa zur Kommunikation mit anderen Mitarbeitern zu verwenden. Werkzeuge sind z.B. Tabellenkalkulationen, mit denen vorhandene Daten beliebig analysiert und bearbeitet werden können, Grafikanwendungen, mit denen Daten beliebig grafisch aufbereitet werden können, die Mail-Software, die die Kommunikation mit anderen Mitarbeitern unterstützt, etc. Aber auch eine FiBu oder eine LBH kann ein speziell zugeschnittenes Werkzeug für einen bestimmten Anwendungsbereich sein, in dem die für den Bereich unveränderlichen Regeln bereits berücksichtigt sind. Der Computer wird zu einem allgegenwärtigen Hilfsmittel bei der Bewältigung der meisten Aufgaben.

Zweiter Lernschritt

Ziel:

Ziel des Lernschrittes ist es, »Unterstützen« und »Ersetzen« als unterschiedliche Entwicklungsvarianten der Zuordnung Mensch - Software einzuführen und das Abbildungsproblem als zentrales Problem der Software-Entwicklung zu begründen.

Inhalt:

Wie arbeitet ein Mensch mit Software-Werkzeugen?

1. Zuordnung von Aufgaben zum Computer und zum Menschen.

- Beispiel Finanzbuchhaltung

  Es können z.B. folgende Teilaufgaben genannt werden:
  Daten eingeben,
  Liste erstellen,
  Anschrift korrigieren,
  Mahnbriefe erstellen.

- Beispiel Textverarbeitung

  Es können z.B. folgende Teilaufgaben genannt werden:
  Text erstellen,

Text formatieren,
Adresse einfügen,
Silbentrennung.

2. »Unterstützen« vs. »Ersetzen« und der Bezug von Software zur individuellen Arbeitsgestaltung.

In der klassischen Dialoganwendung, z.B. dem Finanzbuchhaltungsprogramm im ersten Beispiel, ist eine eindeutige Zuordnung der Tätigkeiten (Mensch oder Computer) möglich. Damit sind auch eine weitgehend genaue Anforderungsdefinition und die Abbildung dieser Anforderungen möglich. Im zweiten Beispiel, eine Software mit Werkzeugcharakter, ist eine genaue Zuordnung auf Computer oder Mensch nicht mehr möglich. Alle Tätigkeiten werden durch den Menschen mit Unterstützung des Computers durchgeführt. Damit gibt es auch keine exakte Anforderungsdefinition für die Informationstechnik, Anforderungen können nicht mehr eindeutig abgebildet werden.

In dieser Form ist die IT eng verknüpft mit der individuellen Arbeitsweise jedes Mitarbeiters und dem Bezug seiner Arbeit zur Arbeit anderer. Der Computer muß also auf die Aufgabe des Einzelnen zugeschnitten sein. Andererseits muß der Mitarbeiter seine Arbeitsweise auf die neuen Werkzeuge einstellen, er muß Erfahrungswissen neu entwickeln.

Dritter Lernschritt

Ziel:

Ziel des Lernschrittes ist es, die Pole »Abbildung« und »Arbeitsgestaltung« als Kern des Qualitätsproblems zu entwickeln und Dimensionen und Kriterien für Software-Qualität zu erarbeiten.

Inhalt:

Bestimmung der Qualität eines Software-Werkzeugs

1. Was ist Software-Qualität?

Traditionell technisch orientierte Qualitätskriterien werden z.B. anhand der Kriterien der DIN 66234 problematisiert.

Software-Qualität ist nach wie vor ein ungelöstes Problem. Technisch orientierte Kriterien sind zwar meßbar, sind aber vor dem Hintergrund des Nutzungskontextes durchaus widersprüchlich zu bewerten. Darüber hinaus decken sie nur einen geringen Teil der Qualität der Software und der Gesamtqualität des neu ent-

standenen Arbeitssystems ab. Kriterien wie Benutzerfreundlichkeit und Effizienz im Arbeitskontext sind eben kontextabhängig und kaum meßbar.

2. Welche Dimensionen hat Software-Qualität?

- technische Qualität
  - Kriterien der technischen Umsetzung vorgefundener Anforderungsdefinitionen
- Qualität der abgebildeten Anforderungen
  - Kriterien der Definition und Abbildung der Anforderungen aus dem Anwendungskontext
- Qualität im Arbeitskontext
  - Kriterien des Einsatzes der Software

3. Die Qualität von Software-Werkzeugen

Mit wachsender Komplexität der Software tritt die technische Umsetzung angesichts des Problems der Anforderungsdefinition in den Hintergrund. Demzufolge ist die technische Qualität einer Software zwar unabdingbar, sie nimmt aber bei der Gesamtbewertung einer Software einen immer geringeren Stellenwert ein.

Qualitativ hochwertige Software zeichnet sich heute v.a. durch die Qualität der in ihr abgebildeten Anforderungen aus. Die Spezifik der in Software-Werkzeugen abgebildeten Anforderungen weist allerdings über sie hinaus. Für Software-Werkzeuge gelten Kriterien wie Flexibilität, Transparenz, eine dem Nutzungskontext angemessene Komplexität, Ergonomie etc. Für sich genommen lassen diese Kriterien keine Aussagen über die Qualität einer Software zu.

Die relevante Qualität eines Software-Werkzeugs erweist sich erst im Arbeitskontext. Die optimale Anpassung des Werkzeugs an die Arbeitsweise und die Nutzung aller Möglichkeiten des Werkzeugs im Arbeitseinsatz sind letztlich entscheidende Qualitätskriterien.

Vierter Lernschritt

Ziel:

Ziel des Lernschrittes ist es, die prozessuale Einheit von »Abbildung« und »Arbeitsgestaltung« als Kernanforderung an den Entwicklungsprozeß zu entwickeln.

Inhalt:

Die Entwicklung von Software-Werkzeugen

Ziel des Entwicklungsprozesses ist es heute, einen Arbeitsablauf als Ganzen effektiver zu gestalten. Um dies zu erreichen, muß einerseits das Werkzeug an die Arbeitsweise angepaßt sein, andererseits muß der Benutzer sich in seiner Arbeitsweise umfassend auf das Werkzeug einstellen. Somit ist der Entwicklungsprozeß ein Problem der Anforderungsdefinition und -abbildung **und** der Arbeitsgestaltung zugleich.

Die IT-Fachkräfte können das zu entwickelnde Werkzeug mit seinen technischen Möglichkeiten nur auf Basis einer möglichst genauen Anforderungsdefinition beschreiben und entwickeln. In Zusammenarbeit mit Benutzern werden Erfahrungswissen und Überlegungen der Benutzer zur Arbeitsgestaltung auf den aktuellen Wissensstand der technischen Möglichkeiten bezogen und so weiterführende Anforderungen an das technische System beschrieben.

Obwohl das DV-Wissen auch der DV-Laien immer mehr zunimmt, und obwohl man in diesem Zusammenhang oft mutmaßt, daß der Anwender von morgen sich seine DV selbst organisiert, wird die Berücksichtigung der technischen Möglichkeiten im Entwicklungsprozeß zu einem zentralen Problem. Moderne IT zeichnet sich (gerade durch ihren modularen Aufbau mit in sich einfachen Einzelteilen) durch eine ungeheure Vielfalt aus. Diese Entwicklungsmöglichkeiten sind nicht einmal für DV-Spezialisten auf Anhieb zu durchschauen.

Für IT-Fachkräfte als Spezialisten für die Informationstechnik besteht die Anforderung, im Entwicklungsprozeß die technischen Möglichkeiten für den Benutzer transparent zu machen, damit er auf dieser Grundlage seine Anforderungen an das technische System und seine Vorstellungen zur Arbeitsgestaltung formulieren kann. Je konkreter die technischen Möglichkeiten bezogen auf den Nutzungskontext beschrieben werden, desto konkreter können die Vorstellungen zur Arbeitsgestaltung entwickelt werden.

AS 3 Die Ebene der Gestaltung von Arbeitsbeziehungen

Erster Lernschritt

Ziel:

Ziel dieses Lernschrittes ist die Bestimmung der IT-Systeme als Teil der Unternehmensinfrastruktur.

Inhalt:

Der Einsatz der IT als Infrastruktur

1. Welche Rolle spielen Informationen heute für ein Unternehmen?

Der zunehmende Konkurrenzdruck auf immer enger werdenden und sich immer rascher wandelnden Märkten verlangt von den Unternehmen steigende Qualität und Diversifikation ihrer Produkte, größere Produktivität, Reaktionsschnelligkeit, Anpassungsfähigkeit und Pünktlichkeit, um prompt auf Kundenbedürfnisse reagieren zu können. Die geforderte Flexibilität ist nur mit einer hohen Verfügbarkeit aller Informationen, jederzeit, überall und in jeder gewünschten Form zu erreichen.

Dies wird in den Unternehmen durch IT-Systeme erreicht, die wie ein Nervensystem das gesamte Unternehmen durchziehen und alle Arbeitsplätze verbinden. So wird an allen Arbeitsplätzen eine schnelle und flexible Informationsverarbeitung möglich. Darüber hinaus werden mit dieser Form betrieblicher Infrastruktur neue Formen von Kommunikation, Kooperation und betrieblichen Strukturen möglich, die wesentlich zur geforderten Flexibilität beitragen.

2. Was ist eine IT-Infrastruktur?

Das IT-System soll heute die Zusammenarbeit aller Mitarbeiter in einem Unternehmen unterstützen. In diesem Sinne muß das IT-System in einem Unternehmen heute eine Infrastruktur sein, in der der Informationsfluß ungehindert funktioniert, über die die Mitarbeiter miteinander kommunizieren und kooperieren.

Mit dem IT-System finden die Mitarbeiter einen »Werkzeugkoffer« (Gesamtheit der Anwendungssoftware), Ressourcen (z.B. Speicher- und Rechenkapazitäten) und ein Angebot von Dienstleistungen (z.B. Verwaltung der Daten, Mail). In ihrer Gesamtheit bezeichnen wir diese Bestandteile als Infrastruktur. In der Infrastruktur spiegeln sich darüber hinaus die Anforderungen der Kommunikation, der Kooperation wider, aber auch betriebliche Strukturen wie die Abteilungsorganisation und hierarchische Verhältnisse. Diese finden wir z.B. in der Verteilung der Ressourcen oder der Zugriffsrechte wieder.

Zweiter Lernschritt

Ziel:

Ziel des Lernschrittes ist es, die Abbildung der Mensch-Maschine-Beziehungen gegen die Unterstützung der Arbeitsbeziehungen zwischen Menschen abzugrenzen und das daraus entstehende Abbildungsproblem als zentrales Problem der Entwicklung einer IT-Infrastruktur einzuführen.

Inhalt:

Die Arbeit der Menschen in einer IT-Infrastruktur.

1. Ein Beispiel

Welche Infrastruktur braucht ein Software-Entwicklungsteam? Welche Bestandteile hat sie?

In diesem Lernschritt kommt es nicht auf Vollständigkeit an. Ziel ist es, sich in das Problem einer Entwicklungsumgebung als spezifischer Infrastruktur hineinzudenken.

Es können z.B. folgende Stichworte fallen:

- CASE-Tool,
- Mail,
- gemeinsame Datenbasis,
- PM-Tool,
- Dokumentenarchiv,
- Konferenzsystem, etc.

Wie wirkt sie auf die Arbeit?

Es können z.B. folgende Stichworte fallen:

- man muß sich an Standards halten,
- die gemeinsame Arbeit wird koordiniert,
- breitere Diskussion von Problemen,
- das Rad wird seltener zweimal erfunden,
- Programme werden besser dokumentiert, etc.

2. Die IT-Infrastruktur und die Gestaltung von Arbeitsbeziehungen

Die Aufgaben, die heute in einem Unternehmen zu bewältigen sind, zeichnen sich durch einen hohen Informationsbedarf und durch intensive Formen der Kooperation aus. Die Informationsbeschaffung erfolgt über das DV-System und über die Kommunikation mit anderen Mitarbeitern. Die Zusammenarbeit der Mitarbeiter realisiert sich über einen gemeinsamen Datenbestand, über die Nutzung gemeinsamer Ressourcen, über Kommunikation. Dies geschieht in definierten Strukturen. Durch die konkrete Form der Infrastruktur werden Möglichkeiten und Grenzen der Arbeitsbeziehungen bestimmt.

Dritter Lernschritt

Ziel:

Ziel des Lernschrittes ist es, die Pole »Abbildung« und »Gestaltung der Arbeitsbeziehungen« als Kern des Qualitätsproblems zu entwickeln.

Inhalt:

Die Qualität einer IT-Infrastruktur

1. Das Beispiel Software-Entwicklungsumgebung

Mit den Teilnehmern wird die Frage besprochen: »Unter welchen Bedingungen kann das System wirklich effizient genutzt werden?«

Es können z.B. folgende Stichworte fallen:

- Transparenz,
- Angepaßtheit an die Arbeitsweise des Teams,
- alle Teammitglieder müssen darauf eingestellt sein,
- alle Teammitglieder nutzen es,
- es muß so flexibel sein, daß es auch in Ausnahmesituationen sinnvoll ist,
- es muß für neue Aufgaben erweitert werden können.

2. Die Qualität einer IT-Infrastruktur

Obwohl nach wie vor die mangelnde Offenheit der Komponenten eines Systems die Systemintegration zu einem erheblichen technischen Problem macht, dessen Lösung für die Entwicklung einer effizienten IT-Infrastruktur eine notwendige Bedingung ist, treten heute technische Kriterien für den Aufbau einer IT-Infrastruktur in den Hintergrund.

Der modulare Aufbau moderner IT-Systeme, die Möglichkeit, Ressourcen beliebig über ein Unternehmen zu verteilen, macht eine Anforderungsdefinition notwendig, auf deren Basis erst die Infrastruktur geplant und umgesetzt werden kann.

Das Beispiel macht deutlich, daß jede Veränderung der Infrastruktur mit einer Veränderung der Arbeitsbeziehungen verknüpft ist. Somit entscheidet die Qualität dieser Anforderungsdefinition letztlich nicht allein über die Qualität der Infrastruktur. Die Qualität realisiert sich erst in konkreten Arbeitsbeziehungen. Erst die Verknüpfung von Kommunikationsbeziehungen, Kooperationsbeziehungen und

betrieblichen Strukturen mit der IT-Infrastruktur läßt letztlich Aussagen über Erfolg oder Mißerfolg der Entwicklungsarbeit zu. Der Gegenpol zu den abgebildeten Anforderungen ist also die Form der Arbeitsbeziehungen.

Vierter Lernschritt

Ziel:

Ziel des Lernschrittes ist es, die prozessuale Einheit von »Abbildung« und »Gestaltung von Arbeitsbeziehungen« als Kernanforderung an den Entwicklungsprozeß zu entwickeln.

Inhalt:

Die Entwicklung einer IT-Infrastruktur

Ziel des Entwicklungsprozesses ist es heute, die Arbeitsbeziehungen zwischen Menschen in den Organisationseinheiten eines Unternehmens effektiver zu gestalten. Um dies zu erreichen, muß einerseits die Infrastruktur an die Form der Arbeitsbeziehungen angepaßt sein, andererseits müssen die Arbeitsbeziehungen auf Basis der neuen Infrastruktur neu entwickelt werden. Somit ist der Entwicklungsprozeß ein Problem der Anforderungsdefinition und -abbildung und der Gestaltung der Arbeitsbeziehungen.

Die IT-Fachkräfte können die Infrastruktur nur auf Basis einer möglichst genauen Anforderungsdefinition beschreiben und entwickeln. In Zusammenarbeit mit Benutzern werden Erfahrungswissen und Überlegungen der Benutzer zur Gestaltung der Arbeitsbeziehungen auf den aktuellen Wissensstand der technischen Möglichkeiten bezogen und so weiterführende Anforderungen an das technische System beschrieben. Auch hier wird die Berücksichtigung der technischen Möglichkeiten im Entwicklungsprozeß zu einem zentralen Problem.

Für IT-Fachkräfte als Spezialisten für die DV steht die Anforderung, im Entwicklungsprozeß die technischen Möglichkeiten für den Benutzer und die Entscheidungsträger transparent zu machen, damit sie auf dieser Grundlage ihre Anforderungen an das technische System und seine Vorstellungen zur Gestaltung der Arbeitsbeziehungen formulieren kann. Je konkreter die technischen Möglichkeiten bezogen auf den Nutzungskontext beschrieben werden, desto konkreter können die Vorstellungen zur Optimierung der Arbeitsbeziehungen entwickelt werden.

AE 3.4 Die aktuelle Entwicklung der DV und der Aufgaben von IT-Fachkräften

AS 1 Die heutige Situation: Ein Übergangsstadium der Informationsverarbeitung

Ziel:

Die aktuelle Situation im DV-Bereich und die eigene Arbeitssituation sollen als Übergangsstadium begriffen werden. Deshalb werden sie in die historische Entwicklung der DV (anknüpfend an AE 2) eingeordnet.

Inhalt:

Im Rückgriff auf die Ergebnisse der AE 2 wird das dort verwendete Raster

- Abbildungsproblematik
- Nutzungskontext
- Technikkontext
- Aufgabenkontext

extrahiert und mit der Leitfrage

»Wie sind vor diesem Hintergrund aktuelle Entwicklungen einzuordnen?«

weiter bearbeitet.

In Auswertung der vorangegangenen Arbeitsschritte der AE 3 werden die drei ersten Elemente des Rasters ausgefüllt:

- Abbildungsproblematik
  Abgebildet werden heute Funktionen, die Tätigkeiten des Menschen unterstützen können. Da immer weniger Tätigkeiten und Arbeitsabläufe abgebildet werden, verschiebt sich das Abbildungsproblem von der Analyse und Abbildung von Arbeitsabläufen hin zur Analyse des Unterstützungsbedarfs und der Abbildung der so gewonnenen Anforderungen. Die Anforderungsdefinition und -abbildung ist so unmittelbar mit der Arbeitsgestaltung verknüpft.
- Nutzungskontext
  Die Arbeit und die Arbeitsorganisation verändern sich, da die Arbeitsteilung zwischen Mensch und Computer weitgehend aufgegeben wird. Der Computer wird zunehmend zur umfassenden Unterstützung integrierter Arbeitsabläufe genutzt.

- Technikkontext
  Der Trend zur umfassenden Unterstützung von Arbeitsabläufen zieht zunehmend den Einsatz flexibler dezentraler Rechnerarchitekturen nach sich. Sie durchziehen das gesamte Unternehmen als spezifische Informationsinfrastruktur und stellen an allen Arbeitsplätzen die notwendigen Software-Werkzeuge zur Verfügung.

AS 2 Die Konsequenzen für die Arbeitssituation

Ziel:

Herausarbeiten der Konsequenzen für die Arbeitssituation aus dem aktuellen Wandel der Situation im DV-Bereich

Inhalt:

Mit den Teilnehmern wird mit den Arbeitsergebnissen der AE 3 das Element »Aufgabenkontext« bearbeitet und inhaltlich gefüllt.

- Aufgabenkontext
  Das Abbildungsproblem ist für Computerfachkräfte nur noch in einem integrierten Prozeß von Anforderungsdefinition und Arbeitsgestaltung zu bewältigen. Diese Integration von Technikgestaltung und Arbeitsgestaltung stellt IT-Fachkräfte vor völlig neue Anforderungen:
  - Technikimmanente Gestaltungskriterien verlieren an Bedeutung.
  - Gestaltungskriterien des Nutzungskontexts können nicht von den Computerfachkräften allein entwickelt werden. Sie sind deshalb auf die Zusammenarbeit mit anderen Mitarbeitern angewiesen.
  - Die direkte Verknüpfung mit der Arbeitsgestaltung macht die Systementwicklung von einem Prozeß abhängig, der DV-fremden Gesetzen unterliegt.

AE 4 Aufgabenverständnis und Kompetenzen von IT-Fachkräften

Vorbemerkung

Ziel dieser Arbeitseinheit ist die Bündelung des bisher Gelernten auf ein **Interpretationsraster** zum ganzheitlichen Verständnis der Aufgaben von IT-Fachkräften. Ausgehend von diesem Interpretationsraster werden die zur Aufgabenerfüllung wesentlichen **Kompetenzen** für IT-Fachkräfte bestimmt. Sie liefern die Orientierungen, die in der auf die konkreten Handlungssituationen abgestimmten AE 5 für die Bereiche »Qualifizierung« und »Beruf« handlungsleitend werden.

Die AE 2 des Qualifizierungsbausteins »Aufgabenverständnis« hat den Blick dafür geöffnet, daß die Aufgaben von IT-Fachkräften einem tiefgreifenden, historischen Veränderungsprozeß unterliegen. Im Mittelpunkt der AE 3 stand die Bearbeitung des Aufgaben- und Anforderungswandels unter der Fragestellung: »Wie erzeuge ich Qualität?«. Durch die Bearbeitung dieser Fragestellung wurde deutlich, daß grundlegend neue Aufgaben und Anforderungen auf IT-Fachkräfte zukommen. Die AE 4 thematisiert und systematisiert diesen Anforderungswandel nun unter dem Blickwinkel der zu seiner Bewältigung notwendigen Kompetenzen.

Das Interpretationsraster zum ganzheitlichen Verständnis der Aufgaben ist durch das Dreieck »Technikgestaltung im Arbeitsgestaltungskontext - Expertenkooperation - Sozialer Prozeß« im Kern beschrieben. Dessen Verständnis ist für die Teilnehmer wichtig, damit sie angesichts einer sich sehr schnell verändernden Berufswelt eigene Orientierungen für ihre berufliche Zukunft/Situation und ihre Ausbildung jeweils neu entwickeln können.

Die Kompetenzen zur Bewältigung dieses Anforderungswandels lassen sich ebenfalls in drei Elementen darstellen, der »Technikgestaltungskompetenz«, der »Kooperationskompetenz« und der »Prozeßkompetenz«.

AE 4.1 Darstellung und Begründung der Ziele und Inhalte der AE 4

AS 1 Zusammenfassung des Standes der Arbeit

Ziel:

Durch die Zusammenfassung der vorangegangenen Arbeitseinheit transparent machen, auf welchem Wissensstand die folgende AE aufbaut.

Inhalt:

Im Einstieg greift der Dozent die im Aufgabenkontext deutlich gewordenen neuen Anforderungen an IT-Fachkräfte in einem kurzen Referat auf. Dabei verdeutlicht er, daß die Verknüpfung von Technik- und Arbeitsgestaltung einen grundlegenden Anforderungswandel für IT-Fachkräfte hervorbringt. Die im zweiten Teil genannten neuen Anforderungen zur Erzeugung von Qualität in der Systementwicklung werden unter diesem Leitgedanken exemplarisch zusammengefaßt.

AS 2 Überleitung zur systematischen Analyse der Anforderungen an IT-Fachkräfte

Ziel:

Die Teilnehmer zur weiteren Auseinandersetzung mit den Anforderungen an IT-Fachkräfte anregen.

Inhalt:

Ausgehend von dem grundlegenden Wandel der Anforderungen stellt der Dozent die Verbindung zur (späteren) beruflichen Situation der Teilnehmer her und leitet die orientierende Frage des dritten Teils ab. Als (spätere) IT-Fachkräfte sehen sich die Teilnehmer mit diesem Anforderungswandel konfrontiert. Daher ist folgende Frage von Bedeutung: »Welche Kompetenzen benötige ich, um diesem Anforderungswandel im Beruf und in der Qualifizierung begegnen zu können?«

Die Beantwortung dieser Frage setzt eine systematische Klärung dessen voraus, was den Anforderungswandel im wesentlichen ausmacht. Daher wird in der AE 4.2 zunächst genau diese Frage bearbeitet: »Was macht den Anforderungswandel im Kern aus?«

Darauf aufbauend läßt sich die Frage nach der allgemeinen Orientierung der Teilnehmer in Beruf und Qualifizierung bearbeiten. Die Leitfrage für diesen Teil lautet: »An welchen Kompetenzen orientiere ich mich angesichts des Anforde-

rungswandels im Beruf bzw. bei der Qualifizierung?« Die AE 4 hat also folgenden Aufbau:

1. Was macht den Anforderungswandel im Kern aus?
2. Worauf orientiere ich mich angesichts des Anforderungswandels im Beruf bzw. bei der Qualifizierung?

In der AE 4.2 wird der herausgearbeitete Wandel der Aufgaben und Anforderungen so zunächst auf ein Interpretationsraster für ein ganzheitliches Aufgabenverständnis zusammengefaßt.

AE 4.2 Interpretationsrahmen für ein ganzheitliches Verständnis der Aufgaben von IT-Fachkräften

AS 1 Vorstellen des allgemeinen Interpretationsrahmens

Ziel:

Kennenlernen des allgemeinen Interpretationsrahmens.

Inhalt:

Um angesichts des Anforderungswandels eine Orientierung in der Qualifizierung oder im Beruf entwickeln zu können, ist ein analytisches Verständnis der Anforderungen notwendig. Daher ist das Ziel dieser AE 4.2 die Erarbeitung eines Interpretationsrasters für dieses analytische Verständnis der Anforderungen.

Mit Bezug auf die im zweiten Teil dargestellten vielfältigen Anforderungen schlägt der Dozent vor, der Frage nach dem Kern des Anforderungswandels anhand folgender Aspekte nachzugehen:

Aspekt 1: Technikgestaltung im Arbeitsgestaltungskontext

Aspekt 2: Systementwicklung in Kooperation mit anderen Experten

Aspekt 3: Systementwicklung in komplexen sozialen Prozessen

Durch einen kurzen Verweis auf Beispiele aus dem zweiten Teil wird die Bedeutung jeder Ebene für die Teilnehmer plausibel gemacht und die Struktur des Vorgehens begründet.

AS 2 Der neue Charakter der Technikgestaltung: Technikgestaltung im Arbeitsgestaltungskontext

Ziel:

Die Teilnehmer sollen erkennen, daß die systematische Durchdringung von Arbeits- und Technikgestaltung die Aufgabe »Technikgestaltung« grundlegend verändert. Technikgestaltung ist ein untrennbarer Teil der Gestaltung der Arbeit und nur noch in diesem Kontext zu bewältigen.

Inhalt:

Im Einstieg in diesen Arbeitsschritt werden die Ergebnisse zum Wechselverhältnis von Arbeits- und Technikgestaltung aus der AE 3 auf folgende Aussage verdichtet: Die Entwicklung moderner IuK-Systeme vollzieht sich immer im Kontext Arbeitsgestaltung. In diesem integrierten Prozeß von Technik- und Arbeitsgestaltung ist das IuK-System sowohl Entwicklungsziel als auch Entwicklungsmedium.

Ausgehend von der Einbettung der Technikgestaltung in die Arbeitsgestaltung vollzieht sich die Systementwicklung im Spannungsfeld zwischen IuK-System als Entwicklungsziel und IuK-System als Entwicklungsmedium. Aus diesen beiden Elementen ergeben sich Anforderungen an IT-Fachkräfte.

Das IuK-System als Entwicklungsziel

Die Anforderungsdefinition als Basis der technischen Gestaltung des IuK-Systems ist vom Prozeß der Arbeitsgestaltung abhängig. Aus der systematischen Verknüpfung des Prozesses der Neugestaltung der Arbeit und der Neugestaltung der Technik resultiert, daß die Zielkonstanz als Prämisse der Systementwicklung (d.h. die theoretische Notwendigkeit, die gesamte Anforderungsdefinition zu Beginn der Systementwicklung abzuschließen) aufgegeben werden muß. Zu zentralen Anforderungen an IT-Fachkräfte im Systementwicklungsprozeß werden somit:

- Definition und Abbildung von Gestaltungsanforderungen aus dem Arbeitskontext,
- die Optimierung des Entwicklungsprozesses statt der ausschließlichen Orientierung an vorab bestimmten optimalen Gestaltungsmerkmalen des Entwicklungsergebnisses.

Das IuK-System als Entwicklungsmedium

Für IT-Fachkräfte als Spezialisten für die Informationstechnik besteht die Anforderung, im Entwicklungsprozeß die technischen Möglichkeiten für den Benutzer

und die Entscheidungsträger transparent zu machen, damit sie auf dieser Grundlage ihre Anforderungen an das technische System und ihre Vorstellungen zur Gestaltung des Arbeitssystems formulieren können. Die IT-Fachkräfte setzen also das IuK-System in seinem jeweiligen Entwicklungsstadium (z.B. Entwurf oder Prototyp) als Entwicklungsmedium im Prozeß der Arbeitsgestaltung ein. Je konkreter die technischen Möglichkeiten in bezug auf die Auswirkungen im Arbeitssystem beschrieben werden, desto konkreter können die Vorstellungen zur Optimierung des Arbeitssystems entwickelt werden.

Mit dem Einsatz des IuK-Systems als Entwicklungsmedium im Prozeß der Arbeitsgestaltung leisten Computerfachkräfte einen eigenständigen Beitrag zur Optimierung dieser Innovationsprozesse. Für die IT-Fachkräfte resultiert aus dieser Funktion die Anforderung, die Optionen bei der Gestaltung der IT in ein Verhältnis zur Gestaltung der Arbeit zu setzen. Das bedeutet im einzelnen:

- **Technik im Kontext »Arbeit« interpretieren und bewerten**

  Anhand eines konkreten Beispiels (z.B. Verfügung über eine Briefvorlage) wird bearbeitet, wie technische Gestaltungsalternativen im Kontext »Arbeit« zu interpretieren sind. Das aus dem Arbeitskontext resultierende Bewertungsmuster für die Gestaltungsalternativen soll konkret nachvollzogen werden.

- **Technik im Interessenbezug interpretieren und bewerten**

  Die Wirkung der Technikgestaltung auf die Arbeit und die Arbeitsgestaltung verbindet sie zwingend mit den Interessen der Beschäftigtengruppen im Unternehmen. Die Alternativen bei der Technikgestaltung wirken sich jeweils unterschiedlich auf die Arbeitsbedingungen aus: sie kommen den Interessen und Bedürfnissen der Gruppen eher entgegen oder laufen ihnen zuwider. Dieser prinzipielle Interessenbezug der Technikgestaltung wird an einem Beispiel in diesem Abschnitt herausgearbeitet.

  Zwei Fragen stehen im Vordergrund:

  - Die Bearbeitung des oben genannten Beispiels unter dem Gesichtspunkt der Identifizierung der Interessen(-konstellation).
  - Die Bearbeitung eines Fallbeispiels unter dem Gesichtspunkt der Wirkung technischer Gestaltungsalternativen auf diese Interessen(-konstellation).

AS 3 Expertenkooperation

Ziel:

Herausarbeiten, wie die Technikgestaltung in Kooperationsprozesse zwischen Experten eingebunden ist und welche Anforderungen hieraus für die IT-Fachkräfte resultieren.

Inhalt:

Einerseits muß die Definition der Anforderungen und deren Abbildung am Erfahrungswissen der Anwender und an der Arbeitsgestaltung orientiert sein. Andererseits ist der Prozeß der Entwicklung des IuK-Systems unmittelbar mit dem Prozeß der Arbeitsgestaltung verknüpft. »Technikgestaltung« ist also nicht mehr im »stillen Kämmerlein« zu erfüllen.

Computerfachkräfte stehen so vor der Anforderung, in der Einheit von Arbeitsgestaltung und Technikgestaltung mit anderen Experten zusammenzuarbeiten. Kooperation mit Benutzern und anderen Experten im Unternehmen ist die einzige Möglichkeit, die Entwicklung des IuK-Systems und die Arbeitsgestaltung sinnvoll und effektiv aufeinander zu beziehen.

Somit basiert Systementwicklung auf der Kooperation mit anderen Experten. Welche Anforderungen an IT-Fachkräfte resultieren daraus?

IT-Fachkräfte als spezielle Experten

Zunächst wird die Expertenidentität der IT-Fachkräfte und ihre Bedeutung in den konkreten Prozessen anhand eines Beispiels erarbeitet. Damit wird die Voraussetzung geschaffen, das Expertentum anderer Gruppen überhaupt zur Kenntnis zu nehmen und in seiner konkreten Funktion vom eigenen Beitrag abzugrenzen.

Die Expertenfunktion der IT-Fachkräfte bezieht sich u.a. auf folgende Aufgaben:

- IT-Fachkräfte realisieren die Gestaltungsanforderungen in Form des technischen Systems.
- IT-Fachkräfte entwickeln technische Gestaltungsalternativen und beziehen sie auf damit verbundene Arbeitsgestaltungsalternativen.
- IT-Fachkräfte unterstützen die Anwender und Vertreter der Fachabteilungen durch die Entwicklung und Präsentation technischer Gestaltungsalternativen bei der Definition von Gestaltungsanforderungen.

Bedeutung und Aufgabe anderer Experten

In diesem Schritt sind die anderen am Prozeß beteiligten Experten und deren Bedeutung für den Systementwicklungsprozeß zu identifizieren. Sie werden anhand ihrer Zuständigkeiten im Arbeitsgestaltungsprozeß z.B. in einem Fallbeispiel identifiziert:

1. Für die Ebene der individuellen Arbeitsgestaltung sind v.a. die späteren Anwender zuständig.
2. Für die Ebene der Gestaltung der Arbeitsbeziehungen sind je nach Entwicklungsziel und -phase verschiedene Gruppen im Unternehmen zuständig. Sie vertreten Gestaltungsinteressen wie
   - Umsetzung der Unternehmensstrategie,
   - Unternehmensorganisation,
   - Arbeitsorganisation u.a..

Zusammenfassung: Anforderungen aus der Kooperation mit anderen Experten

Die Ergebnisse dieses Schritts werden abschließend zusammengefaßt. Um die Diskussion zu strukturieren, faßt der Dozent die Ergebnisse in folgenden Lehrsätzen/Thesen zusammen:

1. Eine effektive Zusammenarbeit mit anderen Experten setzt voraus, daß IT-Fachkräfte das Expertenwissen und die Entwicklungsleistung anderer Experten hinsichtlich der Gestaltung des technischen Systems übersetzen und bewerten.
2. Eine effektive Zusammenarbeit mit anderen Experten setzt voraus, daß die IT-Fachkräfte ihren speziellen Beitrag zur Systementwicklung auf das Expertenwissen und die Entwicklungsleistung der anderen Akteure beziehen.

AS 4 Soziale Prozesse

Ziel:

Systementwicklung als sozialen Prozeß kennzeichnen und die daraus resultierenden Handlungsbedingungen für IT-Fachkräfte in Systementwicklungsprozessen verdeutlichen.

Inhalt:

Die Entwicklung moderner IuK-Systeme ist gekennzeichnet durch

- die mangelnde Zielkonstanz ,
- den Einsatz des IuK-Systems als Medium im Arbeitsgestaltungsprozeß und
- die Kooperation mit anderen Experten.

Damit wird Systementwicklung zu einem komplexen Prozeß, der sich zwischen den Polen

- Technikgestaltung - Arbeitsgestaltung,
- IuK-System als Entwicklungsziel - IuK-System als Entwicklungsmedium bewegt.

Arbeitsgestaltung und die Entwicklung des IuK-Systems, technische Gestaltungskriterien und Gestaltungskriterien aus dem Nutzungskontext müssen in der Zusammenarbeit der verschiedenen Experten immer wieder aufeinander bezogen werden. Die Dynamik dieser Prozesse und damit die Anforderungen an die IT-Fachkräfte werden im wesentlichen durch die unterschiedlichen Gestaltungsinteressen der beteiligten Gruppen bestimmt.

Ebenen sozialer Prozesse bei der Systementwicklung

Im Lehrgespräch werden drei Ebenen sozialer Prozesse bei der Systementwicklung identifiziert:

- innerhalb der **Projektgruppe** als unmittelbar Projektbeteiligte aus den Fachabteilungen, der DV-Abteilung usw.,
- im **Projektzielgebiet**, in den betroffenen Abteilungen bzw. Unternehmensbereichen**,**
- im Unternehmen als **Projektumfeld.**

Soziale Prozesse und Aufgaben von IT-Fachkräften

Die drei Ebenen Projektgruppe, Projektzielgebiet und Projektumfeld lassen sich als drei ineinanderliegende Kreise darstellen. Jede der drei Ebenen bestimmt die Aufgaben der IT-Fachkräfte in besonderer Weise. Dies soll im folgenden aufgezeigt werden.

1. Ebene: Soziale Prozesse in der Projektgruppe

Anhand eines Fallbeispiels wird die Bedeutung der sozialen Prozesse in der Projektgruppe für die Systementwicklung in einem Lehrgespräch entwickelt. Wichtig ist die Identifizierung der Wechselwirkung zwischen den Interessen der Projektbeteiligten einerseits und der Einwirkung der Interessen der verschiede-

nen betroffenen Gruppen auf die Verläufe im Projekt andererseits. Diese Wechselwirkung kann auf folgenden Ebenen betrachtet werden:

- Das Zusammenwirken von IT-Fachkräften und Vertretern der Fachabteilungen sowie anderer Experten.
- Das Zusammenwirken der Vertreter unterschiedlicher Fachabteilungen.
- Das Zusammenwirken der IT-Fachkräfte untereinander.

2. Ebene: Soziale Prozesse im Zielgebiet der Systementwicklung

Anhand eines Fallbeispiels wird die Bedeutung der sozialen Prozesse im Zielgebiet für die Systementwicklung in einem Lehrgespräch entwickelt. Wichtig ist die Identifizierung der Wechselwirkung zwischen der Einwirkung des Systems auf die Arbeit bzw. die Arbeitsorganisation einerseits und der Einwirkung der Interessen der verschiedenen betroffenen Gruppen des Zielgebiets auf die Gestalt des Systems andererseits. Diese Wechselwirkung kann auf folgenden Ebenen betrachtet werden:

- Das IuK-System berührt die Art und Weise des Zusammenwirkens ganzer Unternehmensbereiche bzw. Abteilungen.
- Das IuK-System wirkt sich auf die Arbeitsabläufe, das Stellengefüge sowie das quantitative und qualitative Angebot an Arbeitsplätzen aus.

3. Ebene: Soziale Prozesse im Projektumfeld

Die Bedeutung der sozialen Prozesse im Projektumfeld für die Systementwicklung wird erarbeitet. Wichtig ist die Identifizierung folgender Wirkungsebenen und deren Rückwirkung auf die Aufgaben von IT-Fachkräften:

- Die Unternehmenskultur als übergreifende Orientierung bei der Regulierung von Konflikten, der Entwicklung von Maßstäben und des Umgangs untereinander,
- die Beziehungen zwischen der Unternehmensleitung und der betrieblichen Interessenvertretung (Betriebsrat) als Parameter für die Offenheit im Umgang mit Interessengegensätzen, den Grad der Verrechtlichung und Formalisierung sozialer Prozesse, u.a.

Zusammenfassung: Anforderungen, die aus sozialen Prozessen resultieren

Die Lehrsätze/Thesen lauten:

1. Die anzustrebende Gestalt eines IuK-Systems läßt sich meist nicht in Form eines technischen Optimums im voraus bestimmen.

2. Die konkrete Gestalt eines IuK-Systems ergibt sich vielmehr aus dem Zusammenwirken sehr unterschiedlicher Gruppen mit (häufig) unterschiedlichen Zielen und Interessen.
3. Die Systementwicklungsprozesse sind iterative Prozesse und lassen sich nicht mehr als linear-sequentielle Abläufe steuern.
4. Das Offenlegen technischer Gestaltungsalternativen und deren Wirkungen auf die Gestalt der Arbeit ist die Kernanforderung an IT-Fachkräfte, damit gleiche und widerstreitende Interessen zum Motor der Entwicklungsprozesse werden können.

AS 5 Technikgestaltung als ganzheitliche Aufgabe

Ziel:

Zusammenfassendes Herausarbeiten des ganzheitlichen Aufgabenverständnisses

Inhalt:

Zum Abschluß dieses Kapitels werden die herausgearbeiteten Anforderungen hinsichtlich der Leitfrage: »Was macht den Anforderungswandel im Kern aus?« von den Dozenten in einem kurzen Referat zusammengefaßt.

Leitgedanke der Ausführungen ist die Vermittlung der zwingenden Dreiecksbeziehung zwischen den Aspekten »Technikgestaltung im Arbeitsgestaltungskontext - Kooperation mit Experten - Systementwicklung in sozialen Prozessen«. Das Zusammenwirken dieser drei Aspekte macht den Kern des Anforderungswandels für IT-Fachkräfte aus.

AE 4.3 Kompetenzen von IT-Fachkräften

Vorbemerkung

In diesem Schritt werden die erarbeiteten Kernanforderungen an IT-Fachkräfte auf die zu ihrer Erfüllung erforderlichen Kompetenzen bezogen. Dadurch wird ein Übergang zu einer konkreten Handlungsorientierung der Teilnehmer über die Bearbeitung eines realen Handlungsfeldes in der AE 5 ermöglicht.

AS 1 Überblick über wesentliche Kompetenzen von IT-Fachkräften

Ziel:

Den Teilnehmern einen Überblick über relevante Kompetenzen und ihren Zusammenhang vermitteln.

Inhalt:

Im Einstieg greift der Dozent die wesentlichen Anforderungen an IT-Fachkräfte in einem kurzen Referat auf und verbindet sie mit der Frage nach den erforderlichen Kompetenzen zur Bewältigung dieser Anforderungen.

Die erforderlichen Kompetenzen sollen mit Hilfe der Leitfrage: »An welchen Kompetenzen orientiere ich mich angesichts des Anforderungswandels im Beruf bzw. bei der Qualifizierung?« herausgearbeitet werden.

Das Kurzreferat wird durch eine Graphik unterstützt, welche die drei wesentlichen Anforderungen darstellt.

Analog zu den dargestellten Anforderungen entwickelt der Dozent in diesem Referat die drei zentralen Kernkompetenzen von IT-Fachkräften:

- Technikgestaltungskompetenz,
- Kooperationskompetenz,
- Prozeßkompetenz.

AS 2 Technikgestaltungskompetenz

Ziel:

Herausarbeiten der Technikgestaltungskompetenz als zentraler Kompetenz von IT-Fachkräften.

Inhalt:

Dem Bereich Technikgestaltungskompetenz werden in einem Lehrgespräch einzelne Kompetenzen zugeordnet. Die Vorschläge der Teilnehmer werden mit Hilfe von Karten an der Pinnwand dokumentiert und nach Schwerpunkten sortiert. Folgende Kompetenzen sollten erscheinen:

Technikgestaltungskompetenz

- Systementwicklungsprozesse ohne Zielkonstanz bewältigen;
  - Definition und Abbildung von Gestaltungsanforderungen aus dem Arbeitskontext
  - die Optimierung des Entwicklungsprozesses;
- das IuK-System als Entwicklungsmedium einsetzen;
  - Technik im Kontext »Arbeit« interpretieren und bewerten;
  - Technik im Interessensbezug interpretieren und bewerten.

AS 3 Kooperationskompetenz

Ziel:

Herausarbeiten der Kooperationskompetenz als zentraler Kompetenz von IT-Fachkräften.

Inhalt:

Dem Bereich »Kooperationskompetenz« werden in einem Lehrgespräch einzelne Kompetenzen zugeordnet. Die Vorschläge der Teilnehmer werden mit Hilfe von Karten an der Pinnwand dokumentiert und nach Schwerpunkten sortiert. Folgende Kompetenzen sollten erscheinen:

Kooperationskompetenz

- die eigene Rolle und Aufgabe im Bezug zu anderen Akteuren im Innovationsprozeß bestimmen;
- das Expertenwissen und die Entwicklungsleistung anderer Akteure verstehen und bewerten;
- die eigene Arbeit auf das Expertenwissen und die Entwicklungsleistung der anderen Akteure beziehen.

AS 4 Prozeßkompetenz

Ziel:

Herausarbeiten der Prozeßkompetenz als zentraler Kompetenz von IT-Fachkräften.

Inhalt:

Dem Bereich »Prozeßkompetenz« werden in einem Lehrgespräch einzelne Kompetenzen zugeordnet. Die Vorschläge der Teilnehmer werden mit Hilfe von Karten an der Pinnwand dokumentiert und nach Schwerpunkten sortiert. Folgende Kompetenzen sollten erscheinen:

Prozeßkompetenz

- soziale Prozesse in ihrer Dynamik verstehen;
- eigene Eingriffsmöglichkeiten und Verantwortung erkennen;
- Interessen und Widersprüche als Motor der Entwicklung verstehen und nutzen;
- den Entwicklungsprozeß ohne Zielkonstanz regulierend gestalten;
- Zyklizität und Rückkopplung als Methoden der Entwicklung nutzen.

AS 5 Orientierungen für Qualifizierung und Beruf

Ziel:

Exemplarisches Herausarbeiten der allgemeinen Orientierungswirkung der entwickelten Kompetenzen für konkrete Handlungsfelder.

Inhalt:

An einem Beispiel erläutert der Dozent, welche Orientierungen sich aus den in dieser AE 4 entwickelten inhaltlichen Zusammenhängen zu den Bereichen »Kernaufgaben« und »Kompetenzen« von IT-Fachkräften für die Ebenen »Arbeit im Beruf« und »Qualifizierung« ableiten lassen.

Dieses Beispiel sollte aus dem Handlungsfeld »Projektarbeit in der Software-Entwicklung« entnommen werden und möglichst aus dem Praxisbereich der Teilnehmer stammen bzw. an deren Praxis angelehnt werden.

Zum Schluß dieses Schrittes faßt der Dozent die Ergebnisse zusammen und leitet zum nächsten Arbeitsschritt über.

AE 5 Ausgang »Methoden des Software Engineering«

Gesamtziel der AE 5 (am Beispiel »Unterricht im Fach Software Engineering«)

Diese Arbeitseinheit entwickelt mit den Teilnehmern die konkrete Handlungsorientierung für den Unterricht im Fach »Software Engineering«. Dazu werden ausgehend von der Formulierung relevanter Themen für den Unterricht Absprachen zu deren Umsetzung getroffen.

AS 1 Anforderungen an den Unterricht

Ziel:

Ausgehend von einem ganzheitlichen Verständnis der Aufgaben von IT-Fachkräften werden in diesem Schritt konkrete Anforderungen an Ziel und Inhalt der folgenden Unterrichtseinheit definiert.

Inhalt:

In diesem Arbeitsschritt wird die erarbeitete allgemeine Handlungsorientierung aus der AE 4.3 auf den Unterricht »Software Engineering« konkretisiert.

Bevor den Teilnehmern die Gelegenheit gegeben wird, ihre Anforderungen an die Unterrichtseinheit »Methoden des Software Engineering« zu benennen, stellt der Dozent zunächst eine allgemeinverständliche Definition des »Software Engineering« vor.

Die Teilnehmer sammeln Themen, die sie in der Unterrichtseinheit »Methoden des Software Engineering« behandeln wollen.

AS 2 Umsetzung im Unterricht

Ziel:

Vereinbarungen über die Umsetzung im Unterricht treffen.

Inhalt:

Zum Abschluß wird eine Verabredung getroffen, wie die gesammelten Unterrichtsvorschläge in die Gestaltung der Unterrichtseinheit »Methoden des Software Engineering« einfließen.

Dies kann folgendermaßen gehandhabt werden:

Zu Beginn der Unterrichtseinheit sortiert der Dozent die Karten nach Themengebieten und ordnet sie ihrem Stoffplan zu. Sollte der Dozent nicht an dieser Unterrichtseinheit zum Aufgabenverständnis teilgenommen haben, ist ein Verfahren zu vereinbaren, wie die Vorschläge vermittelt werden.

Andere Themenbereiche:

Weiterhin sollte hier darauf geachtet werden, gegebenenfalls bereits vorher gemachte Vorschläge oder getroffene Vereinbarungen zu anderen Themenschwerpunkten wieder aufzugreifen und Möglichkeiten zu ihrer Umsetzung zu diskutieren.

Das könnten z.B. sein:

- Unterrichtseinheit »Bewerbungstraining und -orientierung« durchführen;
- Stärken und Schwächen der Ausbildung zur Verbesserung des folgenden Ausbildungsgangs zu besprechen;
- Stärken und Schwächen der Praktika zur Verbesserung des folgenden Ausbildungsgangs zu besprechen.

Literatur

[Ackermann/Ulich 1991]
Ackermann, D., Ulich, E.: Software-Ergonomie ´91. Benutzerorientierte Software-Entwicklung, Stuttgart

[Adam 1988]
Adam, E.: Das Subjekt in der Didaktik. Ein Beitrag zur kritischen Reflexion von Paradigmen der Thematisierung von Unterricht, Weinheim

[Aebli 1980]
Aebli, H.: Denken, das Ordnen des Tuns. Band 1, Stuttgart

[Aebli 1981]
Aebli, H.: Denken, das Ordnen des Tuns. Band 2, Stuttgart

[Alheit 1990]
Alheit, P.: Biographizität als Projekt, Bremen

[Anderson/Crocca 1993]
Anderson, W.L., Crocca, W.T.: Engineering practise and codevelopment of product prototypes, in: Communications of the ACM, June 1993, Vol. 36, No.4, S. 49-56

[Andresen u.a. 1986]
Andresen, N.E., Kensing, F., Lassen, M., Ludin, J., Mathiassen, L., Munk-Madsen, A., Sørgaard, P.: Professionel Systemudvikling, Kopenhagen

[Angehrn 1985]
Angehrn, E.: Geschichte und Identität, Berlin u.a.

[Arnold 1988]
Arnold, R.: Was (v)erschließen die Schlüsselqualifikationen?, in: Nuissl/Siebert/Weinberg 1988

[Aschersleben u.a. 1989]
Aschersleben, G., Gstalter, H., Kaiser, F., Strube, V., Zang-Scheucher, B.: Prototyping als Verfahren zur Software-Entwicklung. Literaturanalyse und Expertengespräche, in: Zeitschrift für Arbeitswissenschaft, Nr.43, S. 42-47

[Aschersleben/Zang-Scheucher 1989]
Aschersleben, G., Zang-Scheucher, B.: Der Prozeß der Software-Gestaltung - Eine Bestandsaufnahme in Wissenschaft und Industrie, in: Maaß/Oberquelle 1989, S. 244-253

[Ausubel 1973]
Ausubel, D.P.: Psychologie des Unterrichtens, Weinheim

[Axelrod 1988]
Axelrod, R.M.: Die Evolution der Kooperation, Oldenburg, München

[Bachmann/Möll 1992]
Bachmann, R., Möll, G.: Alles neu ...? Rationalisierung von industriellen Innovationsprozessen. Eine Herausforderung für die industriesoziologische Analyse?, in: Malsch/Mill 1992, S. 241-270

[Badura 1989]
Badura, J.: Die Diskussion um das Konzept "Schlüsselqualifikationen" - Anregungen für die Volkshochschularbeit, in: Meisel u.a. 1989

[Baethge 1989]
Baethge, M.: Jugend, Arbeit und Identität. Lebensperspektiven und Interessenorientierung von Jugendlichen, Opladen

[Baethge 1991]
Baethge, M.: Arbeit, Vergesellschaftung, Identität. Zur zunehmenden normativen Subjektivierung der Arbeit, in: Soziale Welt, Jg.42 (1991), 1, S.6-19

[Baethge/Overbeck 1990]
Baethge, M., Overbeck,H.: Systemische Rationalisierung von Dienstleistungsarbeit und Dienstleistungsbeziehungen: Eine neue Herausforderung für Unternehmen und wissenschaftliche Analyse, in: Rock u.a. 1990, S. 149-175

[Balzert 1989]
Balzert, H.: Die Entwicklung von Software-Systemen. Prinzipien, Methoden, Sprachen, Werkzeuge, in: Reihe Informatik Band 34, Mannheim, Wien, Zürich

[Bardmann 1992]
Bardmann, T.M., Dollhausen, K., Kleinwellfonder, B.: Technik als Parasit der Kommunikation, in: Soziale Welt, H.2 (1992), S. 201-216

[Bateson 1987]
Bateson, G.: Geist und Natur: eine notwendige Einheit, Frankfurt a.M.

[Bauer 1993]
Bauer, F.L.: Software-Engineering- wie es begann, in: Informatik-Spektrum (1993) 16, 259-260

[Baukrowitz u.a. 1992]
Baukrowitz, A., Boes, A., Boß, C., Hütten, U., Jung, U.: Ganzheitliche Arbeitsgestaltungskompetenz. Paradigmenwechsel in der Aus- und Weiterbildung von Informatik-Fachkräften, in: Langenheder/Müller/Schinzel 1992, S. 270-274

[Baukrowitz u.a. 1993]
Baukrowitz, A., Boes, A., Schreiner, G.: Qualifikationssicherung und Qualifikationsentwicklung für Informatik-Fachkräfte, Marburg

[Baumann 1990]
Baumann, R.: Didaktik der Informatik, Stuttgart

[Beck 1993]
Beck, A.: Benutzerpartizipation aus Sicht von SW-Entwicklern und Benutzern. Eine Untersuchung von beteiligungsorientierten SW-Entwicklungsprojekten, in: Rödiger 1993, S.263-274

[Beck u.a. 1980]
Beck, U., Brater, M., Daheim, H.: Soziologie der Arbeit und der Berufe, Hamburg

[Beck/Ilg 1991]
Beck, A., Ilg, R.: Aufgabenorientierte Analyse und Gestaltung mit TASK, in: Frese u.a. 1991, S. 95-106

[Beck/Ziegler 1991]
Beck, A., Ziegler, J.: Methoden und Werkzeuge fü die frühen Phasen der Software-Entwicklung, in: Ackermann/Ulich 1991, S.76-85

[Beck/Janssen 1993]
Beck, A., Janssen, C.: Vorgehen und Methoden für aufgaben- und benutzerangemessene Gestaltung von graphischen Benutzungsschnittstellen, in: Coy u.a. 1993, S. 200-221

[Becker/Paetau 1992]
Becker, B., Paetau, M.: Von der kognitiven zur interaktiven Adäquatheit? Expertensysteme zwischen Substitution und Assistenz menschlicher Problemlösungen, in: Malsch/Mill 1992; S. 133-156

[Beeman 1982]
Beeman, W.: Cultur, Performance and Communication in Iran, Tokyo

[Beland 1988]
Beland: Computerfaszination und Lebensgeschichte, in: Krafft/Ortmann 1988

[Beling 1991]
Beling, J.: Computer am Arbeitsplatz. Spezialbibliographie

[Bender/Graßl 1991]
Bender, C., Graßl, H.: Technik und Interaktion. Zur Theorie und Empirie der Technikforschung, Wiesbaden

[Bernard 1991]
Bernard, F.: Die Entwicklung der Fachdidaktik des technologischen Unterrichts unter dem Aspekt der Informationsverarbeitung, in: Bonz/Lipsmeier 1991

[Berufsförderungszentrum Essen 1988]
Berufsförderungszentrum Essen e.V. (Hg.): Zukunftsinvestition Berufliche Bildung, Bd. 1, Köln

[Beuschel 1993]
Beuschel, W.: Das Fachgebiet "Informatik und Gesellschaft" an amerikanischen Hochschulen - eine Übersicht, in: InfoTech (Jg. 5) Heft 4, Dez.93 - Febr. 94, S. 70-71

[Birke 1992]
Birke, M.: Betriebliche Technikgestaltung und Interessenvertretung als Mikropolitik. Fallstudien zum arbeitspolitischen Umbruch, Wiesbaden

[Bittner/Hesse 1993]
Bittner, U., Hesse, W.: Methoden- und Werkzeugunterstützung von Software-Entwicklern - Ergebnisse einer Praxis-Untersuchung, in: Coy u.a. 1993, S.493-496

[Bjerknes/Ehn/Kyng 1987]
Bjerknes, G., Ehn, P., Kyng, M. (Hrsg.): Computers and Democracy, Avebury u.a.

[Bjerknes/Kautz 1991]
Bjerknes, G., Kautz, K.: Overview - A Key Concept in Computer support for Cooperative Work, in: Friedrich/Rödiger 1991, S. 163-170

[Blomberg 1988]
Blomberg, P.v.: Weiterbildung im Wandel. Konzeptionelle und methodische Innovationen, Hamburg

[Blumberger/Hülsmann 1988]
Blumberger, W., Hülsmann, H. (Hrsg.): Menschen, Zwänge, Denkmaschinen - Zur technologischen Formierung der Gesellschaft II, München

[Boehm 1981]
Boehm, B.W.: Software Engineering Economics, Prentics-Hall

[Boehm 1988]
Boehm, B.W.: A Spiral Model of Software Development and Enhancement, in: IEEE Computer, No. 5 1988, S. 61-72

[Boehm/Papaccio 1988]
Boehm, B.W., Papaccio, P.N.: "Understanding and Controlling Software Costs", in: IEEE Transaction on Software Engineering, Vol. 14, No. 10, Oct.1988

[Boes/Dieckhoff 1990]
Boes, A, Dieckhoff, K.: Neue Führungsorganisation für Arbeitsstrukturen von morgen. Ein Beitrag zur betrieblichen Produktivitätsförderung im Spannungsfeld von Ökonomie, Technik und Arbeit, Berlin, Bielefeld, München

[Boes u.a. 1991]
Boes, A., Knöß, P., Nispel, A.: Personalentwicklung und Weiterbildung in kleinen und mittleren Unternehmen, Eschborn

[Boes u.a. 1993]
Boes, A., Baukrowitz, A., Eckhardt, B.: IT-Fachkräfte in der Identitätskrise, in: Online H. 4/1993, S. 67f.

[Böhle/Milkau 1988]
Böhle, F., Milkau, B.: Vom Handrad zum Bildschirm. Eine Untersuchung zur sinnlichen Wahrnehmung im Arbeitsprozeß, Frankfurt a.M.

[Bojanowski/Brater/Dedering 1991]
Bojanowski, A., Brater, M., Dedering, H.: Qualifizierung als Persönlichkeitsbildung, Frankfurt a.M.

[Böker 1993]
Böker, K.H.: Wie kann ein kooperativer Prozeß zur Gestaltung von PC-Standartsoftware initialisiert und moderiert werden?, in: InfoTech; Jg. 5, Heft 4, Dez. 93 - Febr. 94, S. 66- 69

[Bono 1971]
Bono, E. de: Laterales Denken, Hamburg

[Bonz/Lipsmeier 1991]
Bonz, B., Lipsmeier, A. (Hrsg.): Computer und Berufsbildung, Stuttgart

[Booth 1989]
Booth, P.: An introduction to human-computer interaction, Hove/London/Hillsdale

[Boretty u.a. 1988]
Boretty, R. u.a.: PETRA - Projekt- und transferorientierte Ausbildung, München

[Boß/Roth 1992]
Boß, C., Roth, V.: Die Zukunft der DV-Berufe, in: Sozialverträgliche Technikgestaltung. Materialien und Berichte, Band 31, Opladen

[Bourdieu 1987]
Bourdieu, P.: Die feinen Unterschiede, Frankfurt a.M.

[Bourdieu 1992]
Bourdieu, P.: Die verborgenen Mechanismen der Macht, Hamburg

[Brauer/Brauer 1992]
Brauer, W., Brauer, U.: Wissenschaftliche Herausforderung für die Informatik: Änderungen von Forschungszielen und Denkgewohnheiten, in: Langenheder/Müller/Schinzel 1992

[Brodbeck 1993]
Brodbeck, F.C.: Warum es sinnvoll ist, Kommunikation und Kooperation in Software-Entwicklungsprojekten verstärkt zu kultivieren: Ergebnisse aus einer empirischen Untersuchung, in: Rödiger 1993, S. 237-248

[Brodbeck/Frese]
Brodbeck, F., Frese, M. (Hrsg.): Software-Entwicklung zwischen Anforderung und Realität (im Erscheinen)

[Brodbeck/Sonnentag]
Brodbeck, F.C., Sonnentag, S.: Arbeitsanforderungen und soziale Prozesse in der Software-Entwicklung, in: Coy u.a.1993, S.248-259

[Brödner 1985]
Brödner, P.: Fabrik 2000. Alternative Entwicklungspfade in die Zukunft der Fabrik, Berlin

[Brose/Hildebrandt 1988]
Brose, H.G., Hildebrandt, B. (Hrsg.): Vom Ende des Individuums zur Individualität ohne Ende, Opladen

[Brown 1986]
Brown, J.S.: From cognitive to social ergonomics and beyond, in: Norman/Draper 1986, S.456-486

[Brüggemann 1980]
Brüggemann, B.: Die Utopie der besseren Verständigung. Zur Rekonstruktion des Identitätskonzepts, Frankfurt a.M., New York

[Bruner 1970]
Bruner, J.S.: Der Prozeß der Erziehung, Berlin

[Budde u.a. 1984]
Budde, R., Kuhlenkamp, K., Mathiassen, L., Züllighoven, M. (Hrsg.): Approaches to prototyping, Berlin

[Budde u.a. 1986]
Budde u.a.: Prototypenbau bei der Systementwicklung. Erfahrungen und Probleme, in: Angewandte Informatik, H. 4 und 5, 1986

[Budde u.a. 1992]
Budde, R., Kautz, K., Kuhlenkamp, K., Zülloghoven, H.: Prototyping. An approach to evolutionary system development, New York

[Budde/Züllighoven 1990]
Budde, R., Züllighoven, H.: Software-Werkzeuge in einer Programmierwerkstatt. Ansätze eines hermeneutisch fundierten Werkzeug- und Maschinenbegriffs, München

[Budde/Züllighoven 1992]
Budde, R., Züllighoven, H.: Software tools in a programming workshop, in: Floyd u.a. 1992, S. 252-268

[Bullinger 1985]
Bullinger, H.-J. (Hrsg.): Software-Ergonomie ´85. Mensch-Computer Interaktion. Bericht Nr. 24 des German Chapter of the ACM, Stuttgart

[Bullinger 1990]
Bullinger, H. (Hrsg.): Software-Ergonomie in der Praxis, Berlin

[Bullinger 1993]
Bullinger, H.-J.: Benutzergerechte Gestaltung von Software - eine Herausforderung an den Industriestandort Bundesrepublik Deutschland, in: Coy u.a. 1993, S. 17-38

[Bullinger u.a. 1994]
Bullinger, H.-J., Meitner, H., Krämer, M.: Total Quality Mangement im Büro, in: Office Management 1-2/1994, S. 26-31

[Bullinger/Niemeier 1990]
Bullinger, H.-J., Niemeier, J.: Strategiegeleitete Innovationen - Potentiale des Einsatzes von IuK-Technologien in Büro und Verwaltung, in: Rock u.a. 1990, S. 81-126

[Bungard/Jöns 1993]
Bungard, W., Jöns, I.: CIM- Aus- und Weiterbildung: Seminarkonzepte zum Themenschwerpunkt Organisation, Braunschweig, Wiesbaden

[Bunk/Zedler 1986]
Bunk, G., Zedler, R.: Neue Methoden und Konzepte beruflicher Bildung, Köln

[Burisch 1989]
Burisch, M.: Das Burnout Syndrom, Berlin

[Bødker u.a. 1988]
Bødker, S., Knudson, J.L., Kyng, M., Ehn, P., Madsen, K.H.: Computer support for cooperative design, in: CSCW (1988), S. 377-394

[Capurro 1992]
Capurro, R.: Informatics and hermeneutics, in: Floyd u.a. 1992; S. 363-375

[Capurro 1992a]
Capurro, R.: Die Herausforderung der Informatik für die praktische Philosophie, in: Coy u.a. 1992, S. 343-354

[Carmel/Whitaker/George 1993]
Carmel, E., Whitaker, R.D., George, J.F.: PD and joint application design: A transatlantic comparison, in: Communications of the ACM, June 1993, Vol. 36, No. 4, S.40-48

[Carroll/Mack/Kellogg 1988]
Carroll, J.M., Mack, R.L., Kellogg, W.A.: Interface metaphors and user interface design, in: Helander 1988, S.67-85

[CDI-Stellenmarktanalyse]
CDI GmbH: Stellenmarktanalyse, München

[ComputerInformation 7-8/1992]
Wie die Zauberlehrlinge ... Gespräch mit Joseph Weizenbaum, in: ComputerInformation H. 7-8/1992, S. 3-6

[Corsten/Lempert 1992]
Corsten, M., Lempert, W.: Moralische Dimensionen der Arbeitssphäre. Literaturbericht, Fallstudien und Bedingungsanalysen zum betrieblichen und beruflichen Handeln und Lernen, in: Materialien aus der Bildungsforschung Nr. 42

[Coy 1992]
Coy, W.: Für eine Theorie der Informatik !, in: Coy u.a. 1992, S. 17-32

[Coy 1992a]
Coy, W.: Soft engines - mass-produced software for working people?, in: Floyd u.a. 1992; S. 269-279

[Coy 1992b]
Coy, W.: Informatik - Eine Disziplin im Umbruch?, in: Coy u.a. 1992, S. 1-12

[Coy 1993]
Coy, W.: Innovative Softwaregestaltung - Eine Anregung zum Diskurs, in: Coy u.a. 1993, S.152-156

[Coy u.a. 1992]
Coy, W. u.a.: Sichtweisen der Informatik, Braunschweig, Wiesbaden

[Coy u.a. 1993]
Coy, W., Gorny, P., Kopp, I., Skarpelis, C. (Hrsg.): Menschengerechte Software als Wettbewerbsfaktor: Forschungsansätze und Anwenderergebnisse aus dem Programm "Arbeit und Technik", Stuttgart

[CSCW ´88]
Proceedings ACM conference on computer-supported cooperative work. September 26-29, Portland, Oregon, New York

[CW 12/1994]
Die Zahl der arbeitslosen Profis ist um rund die Hälfte gestiegen, in: Computerwoche 12/1994 vom 25.03.1994

[CW 3/1994]
Endanwender können bei der Arbeitsplatzgestaltung mitreden. EU-Verordnung sorgt für Unruhe in der Branche, in: Computerwoche, 21.Jg. 3/1994

[Czycholl/Ebner 1988]
Chycholl, R., Ebner, H.(Hrsg.): Zur Kritik handlungsorientierter Ansätze in der Didaktik der Wirtschaftslehre, Oldenburg

[Daniel 1993]
Daniel, M.: Computergestütztes Problemlösen und Verbesserung der Wissenskommunikation als humanorientierte Gestaltungsaufgabe, in: Coy u.a. 1993, S. 526-547

[DeMarco 1989]
DeMarco, T.: Software-Projektmanagement, Attenkirchen

[DeMarco/Lister 1991]
DeMarco, T., Lister, T.: Wien wartet auf Dich! Der Faktor Mensch im DV-Management, München, Wien

[Denert 1992]
Denert, E.: Software-Engineering, Berlin, Heidelberg, New York

[Deutschmann 1989]
Deutschmann, C.: Reflexive Verwissenschaftlichung und kultureller "Imperialismus" des Managements, in: Soziale Welt, Jg.40 (1989), 3, S. 374-396

[Dieterich u.a. 1987]
Dietereich, R. u.a.: Psychologische Aspekte der Erwachsenenbildung, Bad Heilbrunn

[DIN 66234 Teil 8]
DIN 66234 Teil 8: Bildschirmarbeitsplätze

[DIN 69 901]
Projektmanagement. Begriffe

[DIN ISO 9000]
DIN ISO 9000: Qualitätsmanagment- und Qualitätssicherungsnormen. Leitfaden zur Auswahl und Anwendung

[Döbert/Nunner-Winkler 1975]
Döbert, R., Nunner-Winkler, G.: Adoleszenzkrise und Identitätsbildung, Frankfurt a.M.

[Domeyer/Grusdat 1993]
Domeyer, V., Grusdat, M.: Lehre in Informatik und Gesellschaft: eine dreistündige Grundstudiumsveranstaltung - der Normalfall?, in: InfoTech (Jg. 5) Heft 4, Dez.93-Febr. 94, S. 59-63

[Dörner 1976]
Dörner, D.: Problemlösen als Informationsverarbeitung, Stuttgart, Berlin, Köln, Mainz

[Dorsch 1982]
Dorsch, F.: Psychologisches Wörterbuch, 10.Aufl., Bern, Stuttgart, Wien

[Dostal 1993]
Dostal, W.: Qualifikation von Fachkräften der Informationstechnik - Grundständige Ausbildung vs. zertifizierte Weiterbildung. Explorativstudie für das BiBB, unveröff.

[Dreitzel 1970]
Dreitzel, H.P. (Ed.): Recent Sociology No.2, New York

[Dreyfuß/Dreyfuß 1987]
Dreyfuß, H.L., Dreyfuß, S.E.: Künstliche Intelligenz. Von den Grenzen der Denkmaschine und dem Wert der Intuition, Reinbeck

[Dross/Lempert 1988]
Dross, C., Lempert, W.: Untersuchungen zur Sozialisation in der Arbeit 1977-1988. Ein Literaturbericht., Berlin

[Dunckel 1989]
Dunckel, H.: Arbeitspsychologische Kriterien zur Beurteilung und Gestaltung von Arbeitsaufgaben im Zusammenhang mit EDV-Systemen, in: Maaß/Oberquelle 1989, S. 69- 79

[Eberleh u.a. 1994]
Eberleh, E., Oberquelle, H., Oppermann, R. (Hrsg.): Einführung in die Software-Ergonomie, Berlin

[Ebner 1988]
Ebner, H.G.: Theoretische Fundierungsversuche handlungsorientierten Lernens, in: Zeitschrift für Berufs- und Wirtschaftspädagogik (84. Jg), H.5, S.387-401

[Ebner 1990]
Ebner, H.G.: Zur subjektiven Aneignung technischer Zivilisation. Skizzen einer pädagogisch-anthropologischen Analyse, in: Reihe: Beiträge zur Berufs- und Wirtschaftspädagogik, hg. v. Czycholl, R., Ebner, H.G., Fachbereich 4 der Univ. Oldenburg, Oldenburg

[Eco 1976]
Eco, U.: Semiotik. Entwurf einer Theorie der Zeichen, München

[Edelstein/Keller 1982a]
Edelstein, W., Keller, M.: Perspektivität und Interpretation. Zur Entwicklung des sozialen Verstehens, in: Edelstein/Keller 1982, S.9-46

[Ehn 1988]
Ehn, P.: Work orientated design of computer artifacts, Stockholm

[Ellis/Whitington 1983]
Ellis, R., Whitington, D. (Ed.): New Directions in Social Skill Training, London, Canberra

[Erikson 1966]
Erikson, E.H.: Identität und Lebenszyklus, Frankfurt a.M.

[Europäische Gemeinschaften - Kommission 1993]
Europäische Gemeinschaften - Kommission: Wachstum, Wettbewerbsfähigkeit, Beschäftigung - Herausforderungen der Gegenwart und Wege ins 21. Jahrhundert - Weißbuch, Luxemburg

[Falck 1991]
Falck, M.: Arbeit in der Organisation. Zur Rolle der Kommunikation als Arbeit in der Arbeit und als Gegenstand technischer Gestaltung, Berlin

[Falck 1992]
Falck, M.: Arbeit in der Organisation, in: Coy u.a. 1992, S.157-170

[Fisher 1990]
Fisher, D.T.: Produktivität und Information Engineering, Braunschweig, Wiesbaden

[Fleischer 1990]
Fleischer, T.: Zur Verbesserung der sozialen Kompetenzen von Lehrern und Schulleitern. Kommunikationskompetenz und Interaktionskultur als Systemanforderung in der Schule, Hohengehren

[Floyd 1981]
Floyd, C.: A Process-oriented Approach to Software Development, in: ACM: The International Computing Symposium '81

[Floyd 1984]
Floyd, C.: A systematic look at prototyping, in: Budde u.a. 1984, S.1-18

[Floyd 1987]
Floyd, C.: Outline of a paradigm change in software engineering, in: Bjerknes/Ehn/Kyng 1987, S. 191-210

[Floyd 1989]
Floyd, C.: Software Entwicklung als Realitätskonstruktion, in: Lippe 1989

[Floyd 1992]
Floyd, C.: Software development and reality construction, in: Floyd u.a. 1992; S. 75-86

[Floyd 1992a]
Floyd, C.: Human questions in computer science, in: Floyd u.a. 1992; S. 11-15

[Floyd 1994]
Floyd, C.: Software-Engineering - und dann?, in: Informatik Spektrum (1994) 17, S. 29-38

[Floyd u.a. 1987]
Floyd u.a.: Scanorama. Methoden, Konzepte und Realisierungsbedingungen von Initiativen alternativer Software-Entwicklung und - Gestaltung in Skandinavien, in: Werkstattbericht Nr. 30 des Landesprogramms "Mensch und Technik - Sozialverträgliche Technikgestaltung", Düsseldorf

[Floyd u.a. 1989]
Floyd, C., Reisin, F.-M., Schmidt, G.: STEPS to Software Development With Users, in: Ghezzi/McDermit 1989

[Floyd u.a. 1994]
Floyd, C., Mehl, R.-M., Reisin, F.-M., Wolf, G.: Projekt PEtS. Partizipative Entwicklung transparenzschaffender Sofware für EDV-gestützte Arbeitsplätze. Endbericht

[Floyd u.a.1992]
Floyd, C., Züllighoven, H., Budde, R., Keil-Slawik, R. (Hrsg.): Software development and reality construction, Berlin u.a.

[Franck 1990]
Franck, R.: Information, in: Sandkühler 1990, S.669-681

[Frese u.a. 1991a]
Frese, M., Kasten, C., Skarpelis, C., Zang-Scheucher, B. (Hrsg.): Software für die Arbeit von morgen. Bilanz und Perspektiven anwendungsorientierter Forschung. Ergänzung zum Tagesband, Krefeld

[Frese u.a.1991]
Frese, M., Kasten, Chr., Skarpelis, C., Zang-Scheucher, B. (Hrsg.): Software für die Arbeit von morgen, Berlin, Heidelberg, New York

[Frese/Ulich/Dzida 1987]
Frese, M., Ulich, E., Dzida, W. (Hrsg.): Psychological issues of human-computer interaction in the work place, Amsterdam

[Frese/Brodbeck 1989]
Frese, M., Brodbeck, F.C.: Computer in Büro und Verwaltung. Psychologisches Wissen für die Praxis, Berlin, Heidelberg

[Friedrich 1980]
Friedrich, J.: Soziologie und Kybernetik. Zum Verhältnis von Naturwissenschaften, Sozialwissenschaften und Systemwissenschaften, Frankfurt a.M.

[Friedrich 1990]
Friedrich, J.: Adaptivität und Adaptierbarkeit informationstechnischer Systeme in der Arbeitswelt - zur Sozialverträglichkeit zweier Paradigmen, in: Reuter 1990, S.178-191

[Friedrich 1992]
Friedrich, J.: Informatik und Gesellschaft in der Hochschullehre, in: Langenheder/Müller/Schinzel 1992, S.259-264

[Friedrich, J.v u.a. 1993]
Friedrich, J.v., Herrmann, Th., Pescheck-Schröder, M., Rolf, A. (Hrsg.): Informatik und Gesellschaft

[Friedrich/Rödiger 1991]
Friedrich, J., Rödiger, K-H. (Hrsg.): Computergestützte Gruppenarbeit (CSCW), Stuttgart

[Friedrich/Rödiger 1991a]
Friedrich, J., Rödiger, K.-H.: Computergestützte Gruppenarbeit - Einleitende Bemerkungen zur ersten deutschen CSCW-Tagung, in: Friedrich/Rödiger 1991, S. 11-16

[Fuchs-Kittowski 1976]
Fuchs-Kittowski, K.: Probleme des Determinismus und der Kybernetik in der molekularen Biologie; 2. Auflage, Jena

[Fuchs-Kittowski 1992]
Fuchs-Kittowski, K.: Theorie der Informatik im Spannungsfeld zwischen formalem Modell und nichtformaler Welt, in: Coy u.a. 1992, S. 71-82

[Fuchs-Kittowski 1992a]
Fuchs-Kittowski, K.: Reflections on the essence of information, in: Floyd u.a. 1992, S. 416-432

[Fuchs-Kittowski/Junker 1993]
Fuchs-Kittowski, K., Junker, H.: Zukünftige Erwartungen an den Gestalter moderner Informationstechnologien, in: InfoTech; Jg. 5, Heft 4, Dez.93 - Febr. 94, S. 42-50

[Fuchs/Hartmann 1993]
Fuchs, P., Hartmann, E.: Partizipative Softwaregestaltung - Methoden und Instrumente zur Beteiligung der Nutzer, in: Coy u.a. 1993, S.361-378

[Furth 1981]
Furth, H.G.: Intelligenz und Erkennen. Die Grundlagen der genetischen Erkenntnistheorie Piagets, Frankfurt a.M.

[Furth 1990]
Furth, H.G.: Wissen als Leidenschaft. Eine Untersuchung über Freud und Piaget, Frankfurt a.M.

[Galbraith 1977]
Galbraith, J.: Organization design, Reding, Mass.

[Geißler 1988]
Geißler, K.: Schlüsselqualifikationen - Ein Schlüssel auch zum Abschließen, in: Nuissl/Siebert/Weinberg 1988

[Gerds 1989]
Gerds, P.: Symbolisierungsfähigkeit und technische Bildung. Die Eliminierung präsentativer Symbole aus der technischen Bildung, Bremen

[Geulen 1989]
Geulen, D.: Das vergesellschaftete Subjekt. Zur Grundlegung der Sozialisationstheorie, Frankfurt a.M.

[Ghezzi/McDemid 1989]
Ghezzi, C., McDermid, J.A. (Eds.): ESEC '89 - 2nd European Software Engineering Conference, University of Warwick, Coventry, in: Lecture Notes in Computer Science No. 387, New York

[Goffmann 1971]
Goffmann, E.: Interaktionsrituale. Über Verhalten in direkter Kommunikation, Frankfurt a.M.

[Gould 1988]
Gould, J.D.: How to design usable systems, in: Helander 1988, S. 757-789

[Gould/Lewis 1985]
Gould, J.D., Lewis, C.: Designing for usability - key principles and what designers think., in: Proceedings of CHI´83 conference on human factors in computing systems, S.50-53, Boston

[Grennbaum/Kyng 1991]
Greenbaum, J., Kyng, M.: Design at work, Hillsdale/N.J.

[Greutmann 1993]
Greutmann, T.: Datenmodellierung und aufgabengerechte Dialoge: ein Synchronisationsproblem, in: Rödiger 1993, S. 99-110

[Grønbaeck u.a. 1993]
Grönbaeck, K., Kyng, M., Mogensen, P.: CSCW challenges: Cooperative design in engineering projekts, in: Communications of the ACM, Vol 36, No 4, June 1993, S.67-77

[Gumperz/Hymes 1986]
Gumperz, J., Hymes, D.: Directions in Sociolinguistics, New York

[Haaks 1991]
Haaks, D.: Anpassbare Informationssysteme - Basis für aufgabenorientierte Systemgestaltung und Funktionalität, in: Ackermann/Ulich 1991, S. 291-300

[Haaks 1992]
Haaks, D.: Anpaßbare Informationssysteme. Auf dem Weg zu aufgaben- und benutzerorientierter Systemgestaltung und Funktionalität, Göttingen, Stuttgart

[Habermas 1970]
Habermas, J.: Toward a theory of communicative competence, in: Dreitzel 1970

[Habermas 1971]
Habermas, J.: Vorbereitende Bemerkungen zu einer Theorie der kommunikativen Kompetenz, in: Habermas/Luhmann 1971

[Habermas 1973]
Habermas, J.: Stichworte zu einer Theorie der Sozialisation, in: Habermas 1973a

[Habermas 1973a]
Habermas, J.: Kultur und Kritik, Frankfurt a.M.

[Habermas 1988]
Habermas, J.: Theorie des kommunikativen Handelns. Band 1. Handlungsrationalität und gesellschaftliche Rationalisierung, Frankfurt a. M.

[Habermas 1988a]
Habermas, J.: Theorie des kommunikativen Handelns. Band 2. Zur Kritik der funktionalistischen Vernunft, Frankfurt a. M.

[Habermas/Luhmann 1971]
Habermas, J., Luhmann, N.: Theorie der Gesellschaft oder Sozialtechnologie - Was leistet die Systemforschung?, Frankfurt a.M.

[Hack 1987]
Hack, L.: Wie man die Wirklichkeit auf/um den Begriff bringt. Zur handlungstheoretischen Erzeugung ungeahnter Autonomiespielräume, in: Malsch/Seltz 1987

[Hacker 1978]
Hacker, W.: Allgemeine Arbeits- und Ingenieurpsychologie. Psychische Struktur und Regulation von Arbeitstätigkeiten, Berlin; 2. überarb. Aufl.

[Hacker 1986]
Hacker, W.: Arbeitspsychologie. Psychische Regulaton von Arbeitstätigkeiten, Bern

[Hacker 1987]
Hacker, W.: Softwaregestaltung als Arbeitsgestaltung, in: Fähnrich 1987, S. 29-42

[Hacker 1992]
Hacker, W.: Expertenkönnen. Erkennen und Vermitteln, in: Arbeit und Technik: Praxisorientierte Beiträge aus Psychologie und Informatik, Band II, Göttingen, Stuttgart

[Hacker/Müller-Rudolf/Schwarzer-Schönfelder 1989]
Hacker, W., Müller-Rudolf, E., Schwarzer-Schönfelder, E.: Hilfsmittel für die kooperative Aufgabenanalyse - Eine Vorausetzung aufgabenorientierter Systemgestaltung, in: Maaß/Oberquelle 1989

[Halskov Madsen/Aiken 1993]
Halskov Madsen, K., Aiken, P.H.: Experiences using cooperative interactive storyboard prototyping, in: Communication of the ACM, June 1993, Vol. 36, No. 4

[Hamborg/Schweppenhäußer 1993]
Hamborg, K.-C., Schweppenhäußer, A.: Zur Bedeutung psychologischer Arbeits- und Aufgabenanalyse für die Softwaregestaltung, in: Rödiger 1993, S. 227-236

[Hamborg/Schweppenhäußer 1991]
Hamborg, K.-C., Schweppenhäußer, A.: Expertise: Arbeits- und Softwaregestaltung, in: Hans-Böckler-Stiftung (Hrsg.): Manuskripte 64

[Hammer/Pordesch 1993]
Hammer, V., Pordesch, U.: "Sozialorienierte Gestaltung von Informationstechnik" als Informatik-Nebenfach, in: InfoTech (Jg. 5) Heft 4, Dez. 1993 - Febr. 1994, S. 51-58

[Hampe-Neteler/Rödiger 1992]
Hampe-Neteler, W., Rödiger, K.-H.: Software-Ergonomie. Verfahren der Evaluierung und Standards zur Entwicklung von Benutzeroberflächen. Bericht Nr. 2/1992, Fachbereich Mathematik/Informatik, Universität Bremen

[Hansen/Haas 1988]
Hansen, W.J., Haas, C.: Reading and Writing with Computers: A Framework for Explaining Differences in Performance, in: Comm. of the ACM, Vol. 31, Sept. 1988

[Hartmann 1992]
Hartmann, C.: Technische Interaktionskontexte. Aspekte einer sozialwissenschaftlichen Theorie der Mensch-Computer-Interaktion, Wiesbaden

[Hartmann 1993]
Hartmann, M.: Informatiker zwischen Professionalisierung und Proletarisierung, in: Soziale Welt, Jg. 44 (1993), H.3, S. 392-419

[Hauf-Tulodziecki 1993]
Hauf-Tulodziecki, A.: Von der Informatik zur informationstechnischen Bildung. Zur Curriculardebatte in der Schule, in: InfoTech (Jg. 5) Heft 4, Dez. 1993 - Febr. 1994, S. 34-41

[Heeg 1988]
Heeg, F.J.: Empirische Software-Ergonomie. Zur Gestaltung benutzergerechter Mensch/Computer Dialoge

[Heidegger 1987]
Heidegger, G.: Dialektik und Bildung. Widersprüchliche Strukturierungen in Kognition und Emotion, Weinheim/München

[Heidegger 1988]
Heidegger, G.: Offene Berufsausbildung und Chancengleichheit, in: Berufsförderungszentrum Essen 1988

[Heidegger u.a. 1991]
Heidegger u.a.: Berufsbilder 2000. Soziale Gestaltung von Arbeit, Technik und Bildung, Opladen

[Heilmann 1981]
Heilmann, H.: Modelle und Methoden der Benutzermitwirkung in Mensch-Computer-Systemen, Wiesbaden

[Helander 1988]
Helander, M. (Hrsg.): Handbook of human-computer interaction, Amsterdam

[Heller 1992]
Heller, K.A.: Zur Rolle der Kreativität in Wissenschaft und Technik, in: Psychologie und Erziehung im Unterricht, 39. Jg. (1992), H.2

[Henderson 1993]
Henderson, A.: Views of work, the foundation of architecture, in: Rödiger 1993, S.31-50

[Herrmann 1991]
Herrmann, T.: Dispositionsspielräume bei der Kooperation mit Hilfe vernetzter Systeme, in: Frese u.a. 1991, S. 57-68

[Herrmann 1991a]
Herrmann, T.: Die Bedeutung menschlicher Kommunikation für die Kooperation und für die Gestaltung computerunterstützter Gruppenarbeit, in: Oberquelle 1991a, S. 63-78

[Heß u.a. 1991]
Heß, A., Hundsinger, H., Walz, W.: Neue Technologien in den Lehrplänen der gewerblich-technischen Schulen, in: Bonz/Lipsmeier 1991

[Hesse 1991]
Hesse, W.: Neues Denken in der Softwarewelt. unveröff. Manuskript, Marburg

[Hesse 1994]
Hesse, W., Barkow, G., von Braun, H., Kittlaus, H.B., Scheschonk, G.: Terminologie der Softwaretechnik. Ein Begriffssystem für die Analyse und Modellierung von Anwendungssystemen. Teil 1: Begriffssystematik und Grundbegriffe, in: Informatik Spektrum (1994) 17, S. 39-47

[Hesse u.a. 1994a]
Hesse, W., Barkow, G., von Braun, H., Kittlaus, H.B., Scheschonk, G.: Terminologie der Softwaretechnik. Ein Begriffssystem für die Analyse und Modellierung von Anwendungssystemen. Teil 2: Tätigkeits- und ergebnisbezogene Elemente, in: Informatik Spektrum (1994) 17, S. 96-105

[Heydorn 1980]
Heydorn, H.J.: Abstand und Nähe. Wilhelm von Humboldt, in: Heydorn 1980a

[Heydorn 1980a]
Heydorn, H.J.: Ungleichheit für alle. Zur Neufassung des Bildungsbegriffs, Bildungstheoretische Schriften Bd.1, Frankfurt a.M.

[Hildebrandt/Seltz 1987]
Hildebrandt, E., Seltz, R.: Managementstrategien und Kontrolle, Berlin

[Hoff 1986]
Hoff, E.-H.: Arbeit, Freizeit und Persönlichkeit, Bern, Stuttgart, Toronto

[Hoff/Lappe/Lempert]
Hoff, E.-H., Lappe, L., Lempert, W. (Hrsg.): Arbeitsbiographie und Persönlichkeitsentwicklung, Bern, Stuttgart, Toronto

[Holz auf der Heide 1993]
Holz auf der Heide, B.: PROTOS- Methoden zur Entwicklung und Bewertung von Prototypen für Dialogsysteme. Bericht Nr.19 aus dem Lehrstuhl für Psychologie der TU München, München

[Holzkamp 1986]
Holzkamp, K.: Sinnliche Erkenntnis. Historischer Ursprung und gesellschaftliche Funktion der Wahrnehmung; 5.Auflage, Frankfurt a. M.

[Höpfner 1991]
Höpfner, H.-D.: Entwicklung selbständigen Handelns in der beruflichen Aus- und Weiterbildung. Ein auf der Theorie der Handlungsregulation begründetes didaktisches Modell, Berlin

[Hoyos/Holz auf der Heide/Ortlieb 1993]
Hoyos, Carl Graf, Holz auf der Heide, B., Ortlieb, S.: Eine iterative Software-Entwicklungsstrategie mit gezielter Benutzerbeteiligung und systematischer Evaluation der Benutzerfreundlichkeit, in: Coy u.a. 1993, S. 497-525

[Huber 1987]
Huber, G.L.: Kooperatives Lernen: Theoretische und praktische Herausforderung für die pädagogische Psychologie, in: Zeitschrift für Entwicklungspsychologie und Pädagogische Psychologie, Band XIX, H. 4, S. 340-362, 1987

[Huebner u.a. 1988]
Huebner, M., Krafft, A., Ortmann, G.: Auf dem Rücken fliegen. Thrills am Computer, in: Krafft/Ortmann 1988

[Hutchins/Hollan/Norman 1986]
Hutchins, E.L., Hollan, J.D., Norman, D.A.: Direct manipulation interfaces, in: Norman/Draper 1986, S. 125-152

[Hymes1979]
Hymes, D.: Soziolinguistik. Zur Ethnographie der Kommunikation, Frankfurt a.M.

[IDC 1989]
Infomations Informatic 89 - Der deutsche Markt für Software und Sources 1988-1994

[Informatik und Verantwortung 1992]
Stellungnahme des Fachbereichs "Informatik und Gesellschaft" der Gesellschaft für Informatik, in: Coy u.a. 1992, S. 311-326

[ISO 9241]
ISO 9241 Part 10: Ergonomic requirements for office work with visual display terminals. Dialog principles

[Joas 1980]
Joas, H.: Praktische Intersubjektivität. Die Entwicklung des Werkes von G.H. Mead, Frankfurt a.M.

[Johnson 1988]
Johnson, G.: Organisationskrise und Gruppenregression. Anmerkungen zu Kidders »Die Seele einer neuen Maschine«, in: Krafft/Ortmann 1988

[Johnson/Johnson 1979]
Johnson, D.W., Johnson, R.T.: Conflict in the classroom, controversy and learning, in: Review of Educational Research, No 49, 1979

[Kadritzke 1993]
Kadritzke, U.: Ein neuer Expertentyp? Technische Dienstleistungsarbeit zwischen Marktorientierung und Professionsbezug, in: PROKLA, 23. Jg. (1993) H.2

[Kamp 1993]
Kamp, L. (Hrsg.): Der Mensch im computergesteuerten Unternehmen. Möglichkeiten und Grenzen der Gestaltung vernetzter Produktionssysteme, Köln

[Katzenberg/Piela 1993]
Katzenberg, B., Piela, P.: Work language analysis and the naming problem, in: Communicaitons of the ACM, June 1993, Vol.36, No.4

[Keil-Slawik 1989]
Keil-Slawik, R.: Systemgestaltung mit Aufgabennetzen, in: Maaß/Oberquelle 1989, S. 123-133

[Keil-Slawik 1992]
Keil-Slawik, R.: Artifacts in software-design, in: Floyd u.a. 1992; S. 168-188

[Kell 1991]
Kell, A.: Zur Grundlegung einer berufsbezogenen informationstechnischen Bildung: Probleme und Konzepte informationstechnischer Grundbildung, in: Bonz/Lipsmeier 1991

[Keller 1982]
Keller, M.: Die soziale Konstitution des sozialen Verstehens: Universelle und differentielle Aspekte, in: Edelstein/Keller 1982, S. 266-288

[Kensing/Munk-Madsen 1993]
Kensing, F., Munk-Madsen, A.: PD: Structure in the toolbox, in: Communications of the ACM, Vol 36, No.4, June 1993, S. 78-85

[Kirchhof 1991]
Kirchhof, R.: Handlungsorientiertes Lernen, in: Bonz/Lipsmeier 1991

[Kißler 1990]
Kißler, L. (Hg.): Partizipation und Kompetenz. Beiträge aus der empirischen Forschung, Opladen

[Klaus 1972]
Klaus, G.: Semiotik und Erkenntnistheorie, 3.Aufl., Berlin

[Kling 1987]
Kling, R.: Computerization as an ongoing social and political process, in: Bjerknes/Ehn/Kyng 1987, S.117-136

[Kling 1991]
Kling, R.: Cooperation, coordination and control in computer supported work, in: Communications of the ACM, Dec. 1991, Vol 34, No 12

[Klotz 1993]
Klotz, U.: Software als Wettbewerbsfaktor - Perspektiven von Technologiepolitik und Informatikindustrie vor dem Hintergrund der aktuellen Standortdiskussion, in: Coy u.a. 1993, S. 100-130

[Kohlstruck 1990]
Kohlstruck, M.: Person-Subjektivität-Identität, Bremen

[Kornwachs 1993]
Kornwachs, K.: Information und Kommunikation. Zur menschengerechen Technikgestaltung, Berlin

[Koslowski 1988]
Koslowski, K.: Partizipative Systementwicklung und Software-Engineering, Opladen

[Krafft/Ortmann 1988]
Krafft, A., Ortmann, G. (Hrsg.): Computer und Psyche, Frankfurt a. M.

[Krappmann 1968]
Krappmann, L.: Soziologische Dimensionen der Identität. Strukturelle Bedingungen für die Teilnahme an Interaktionsprozessen, Stuttgart

[Krause 1992]
Krause, B.: Kognitives Lernen - Ansätze und experimentelle Befunde, in: Zeitschrift für Psychologie, H. 200 (1992)

[Kubicek/Höller 1991]
Kubicek, H., Höller, H.: Das Organisationskonzept teilautonomer Arbeitsgruppen als Leitbild für die Gestaltung von Groupware-Systemen, in: Oberquelle 1991a, S. 149-174

[Kuhn 1967]
Kuhn, T.S.: Die Struktur wissenschaftlicher Revolutionen, Frankfurt a.M.

[Kuhnt 1992]
Kuhnt, B.: Zukunftswerkstatt als Methode der Arbeitsanalyse und -gestaltung. Motivation zur Partizipation, in: Langenheder/Müller/Schinzel 1992, S.248-252

[Kyng 1991]
Kyng, M.: Designing for cooperation: Cooperating in Design, in: Communications of the ACM, Dec. 1991, Vol 34, No. 12

[Langenheder/Müller/Schinzel 1992]
Langenheder, W., Müller, G., Schinzel, B. (Hrsg.): Informatik cui bono?, Berlin, Heidelberg, New York

[Laurel 1986]
Laurel B.K.: Interface as mimesis, in: Norman/Draper 1986

[Lauter 1987]
Lauter, B.: Software-Ergonomie in der Praxis. Software anwenderfreundlich schreiben, München, Wien

[Laux 1981]
Laux, G. (Hrsg.): Lexikon der Kybernetik, Berlin

[Lehmann 1980]
Lehmann, M.M.: Programms, Life Cycles and Laws of Software Evolution, in: IEEE Proceedings 68 (1980) No. 9, S. 1060-1076

[Lempert 1977]
Lempert, W.: Untersuchungen zum Sozialisationspotential gesellschaftlicher Arbeit, Berlin

[Lempert 1986]
Lempert, W.: Sozialisation in der betrieblichen Ausbildung. Der Beitrag zur Entwicklung sozialer Orientierungen im Spiegel neuerer Längsschnittuntersuchungen, in: Thomas/Elstermann 1986

[Lempert/Corsten 1992]
Lempert, W., Corsten, M.: Moralische Dimension der Arbeitssphäre, Literaturbericht, Fallstudien und Bedingungsanalysen zum beruflichen Handeln und Lernen, Berlin

[Lempert/Franzke 1976]
Lempert, W., Franzke, R.: Die Berufsbeziehung, München

[Leontjew 1973]
Leontjew, A.N.: Probleme der Entwicklung des Psychischen, Frankfurt a.M.

[Leontjew 1979]
Leontjew, A.N.: Tätigkeit, Bewußtsein, Persönlichkeit, Berlin

[Leontjew 1980]
Leontjew, A.N.: Tätigkeit und Kommunikation, in: Sowjetwissenschaft, Gesellschaftswissenschaftliche Beiträge Heft 5, S. 522-535 (1980)

[Lewis 1986]
Lewis, C.: Understanding what´s happening in system interaction, in: Norman/Draper 1986, S.171-186

[Liedtke 1985]
Liedtke, B.: Towards a Change of Perspektive in Systems Development, in: Working Conference on Development an Use of computer-based Systems and Tools - in the context of the Democratisation of Work

[Lippe 1989]
Lippe, W.-M. (Hrsg.): Software-Entwicklung, Berlin

[Lipsmeier 1991]
Lipsmeier, A.: Ganzheitlichkeit, Handlungsorientierung und Schlüsselqualifikation - über den berufspädagogischen Gehalt der neuen Zielgrößen für die berufliche Bildung im Kontext der neuen Technologien, in: Bonz/Lipsmeier 1991

[Lischka 1989a]
Lischka, C.: Konnektionismus, KI und Informatik, in: KI (1989) H.3

[Lischka/Diederich 1987]
Lischka, C., Diederich, J.: Gegenstand und Methode der Kognitionswissenschaft, in: GMD-Spiegel, Heft 2/3 1987, S.21-32

[Lisop 1988]
Lisop, I.: Schlüsselqualifikation - Zukunftsbewältigung ohne Sinn und Verstand, in: Nuissl/Siebert/Weinberg 1988

[Loch 1981]
Loch, W.: Anfänge der Erziehung - Zwei Kapitel aus einem verdrängten Curriculum, in: Maurer 1981

[Lück/Boes 1990]
Lück,R., Boes,A.: Wege zu einer frauenfreundlichen Personalpolitik. RKW-Materialien für kleine und mittlere Unternehmen, Eschborn

[Ludewig/Schneider 1992]
Ludewig, J., Schneider, K.: Software Engineering im Unterricht der Hochschulen - Studienführer Software-Engineering, Stuttgart

[Luft 1992]
Luft, A.L.: "Wissen" und "Information" bei einer Sichtweise der Informatik als Wissenstechnik, in: Coy u.a. 1992

[Luhmann 1972]
Luhmann, N.: Funktion und Folgen formaler Organisation, 2.Aufl. Berlin

[Luhmann 1982]
Luhmann, N.: Autopoiesis, Handlung und kommunikative Verständigung, in: Zeitschrift für Soziologie, Jg. 11 (1982), S. 366-379

[Luhmann 1984]
Luhmann, N.: Soziale Systeme, Frankfurt a.M.

[Luhmann/Schorr 1992]
Luhmann, N., Schorr, K.E.: Zwischen Absicht und Person. Fragen an die Pädagogik, Frankfurt a.M.

[Lukat 1985]
Lukat, A.: Organisation und Methodik für Projekte zum Einsatz von Informationstechnik, in: Arbeitspapiere der GMD 163,, Sankt Augustin

[Lüscher 1968]
Lüscher, K.: Der Prozeß der beruflichen Sozialisation, Stuttgart

[Maaß 1988]
Maaß, J.: In-Formation durch Information - Ausbildung in der technologischen Formation aus systemtheoretischer Sicht, in: Blumberger/Hülsmann 1988

[Maaß 1991]
Maaß, S.: Computergestützte Kommunikation und Kooperation, in: Oberquelle 1991a, S. 11-36

[Maaß 1993]
Maaß, S.: Software-Ergonomie. Benutzer- und aufgabenorientierte Systemgestaltung, in: Informatik-Spektrum (1993) 16, S. 191-205

[Maaß u.a. 1993]
Maaß, S. u.a.: Software-Ergonomie- Ausbildung in Informatik-Studiengängen bundesdeutscher Universitäten, in: Informatik-Spektrum (1993) 16, S.25-38

[Maaß/Oberquelle 1989]
Maaß, S., Oberquelle, H. (Hrsg.): Software-Ergonomie ´89. Aufgabenorientierte Systemgestaltung und Funktionalität, Stuttgart

[Maaß/Oberquelle 1992]
Maaß, S., Oberquelle, H.: Perspectives and metaphors for human-computer interaction, in: Floyd u.a. 1992; S. 233-251

[Mader 1985]
Mader: Interview, in: Siebert 1985

[Malsch 1987]
Malsch, T.: Die Informatisierung des betrieblichen Erfahrungswissens und der "Imperialismus der instrumentellen Vernunft", in: Zeitschrift für Soziologie Jg. 16, H. 2, April 1987, S. 77-91

[Malsch 1987a]
Malsch, T.: " Neue Produktionskonzepte" zwischen Rationalität und Rationalisierung - Mit Kern und Schumann auf Paradigmensuche, in: Malsch/Seltz 1987, S.53-80

[Malsch 1992]
Malsch, T.: Vom schwierigen Umgang der Realität mit ihren Modellen. Künstliche Intelligenz zwischen Validität und Viabilität, in: Malsch/Mill 1992; S. 157-184

[Malsch/Mill 1992]
Malsch, T., Mill, U. (Hrsg.): ArBYTE, Berlin

[Malsch/Seltz 1987]
Malsch, T., Seltz, R.: Die neuen Produktionskonzepte auf dem Prüfstand, Berlin

[Mambrey 1986]
Mambrey, P.: Praxis und Perspektiven partizipativer Systementwicklung, in: Rolf 1986, S. 146-157

[Mandl/Sarda 1988]
Mandl, H., Sparda, H. (Hrsg.): Wissenspsychologie, München, Weinheim

[March/Simon 1976]
March, J.G., Simon, H.: Organisation und Individuum. Menschliches Verhalten in Organisationen, Wiesbaden

[Marx MEW 23]
Marx, K.: Das Kapital. Kritik der politischen Ökonomie. Erster Band, Berlin

[Maurer 1981]
Maurer, F.: Lebensgeschichte und Identität. Beiträge zu einer Geographischen Anthropologie, Frankfurt a.M.

[Mayer/Schumm 1981]
Mayer, E., Schumm, W.: Betriebliche Ausbildung und gesellschaftliches Bewußtsein. Die berufliche Sozialisation Jugendlicher, Frankfurt a.M.

[Mead 1968]
Mead, G.H.: Geist, Identität und Gesellschaft, Frankfurt a.M.

[Meggle 1993]
Meggle, G. (Hg.): Handlung-Kommunikation-Bedeutung, Frankfurt a.M.

[Mehl u.a. 1993]
Mehl, W.-M., Reisin, F.-M., Wolf, G.: PEtS-Seminar Management und Methoden partizipativer Softwareprojekte, in: Werkstattberichte 107 des Landesprogramms "Mensch und Technik - Sozialverträgliche Technikgestaltung", Düsseldorf

[Meisel u.a. 1989]
Meisel u.a.: Schlüsselqualifikationen in der Diskussion, Frankfurt a.M.

[Mellin-Olsen 1981]
Mellin-Olsen, S.: Instrumentalism as an educational concept, in: Educational Studies in Mathematics, 12, S. 351-367

[Merten 1977]
Merten, K.: Kommunikation. Eine Begriffs- und Prozeßanalyse, Opladen

[Miller 1986]
Miller, M.: Kollektive Lernprozesse. Studien zur Grundlegung einer soziologischen Lerntheorie, Frankfurt a.M.

[Miyata/Norman 1986]
Miyata, Y., Norman, D.A.: Psychological issues in support of multiple activities, in: Norman/Draper 1986

[Mollenhauer 1976]
Mollenhauer, K.: Theorien zum Erziehungsprozeß, München

[Molzberger 1983]
Molzberger, P.: Und Programmieren ist doch eine Kunst, in: Molzberger/Schelle 1983

[Molzberger/Schelle 1983]
Molzberger, P., Schelle, H. (Hg.): Psychologische Aspekte der Softwareentwicklung, München

[Morel u.a. 1989]
Morel, J. u.a.: Soziologische Theorie, München, Wien

[Müller 1988]
Müller, E.: Software-Engineering als kreativer Prozeß, Bern, Frankfurt a.M.

[Müller 1989]
Müller, E.W.: Computer im Curriculum - technische Innovation und pädagogische Reflexion, Kiel

[Müller-Holz auf der Heide/Hacker 1991]
Müller-Holz auf der Heide, B., Hacker, S.: Prototyping in einem Designteam: Vorgehen und Erfahrungen bei einer Software-Entwicklung unter Benutzerbeteiligung, in: Ackermann/Ulich 1991, S. 108-118

[Mumford/Welter 1984]
Mumford, E., Welter, G.: Benutzerbeteiligung bei der Entwicklung von Computersystemen, Berlin

[Nake 1986]
Nake, F.: Die Verdoppelung des Werkzeuges, in: Rolf 1986, S. 43-52

[Nake 1992]
Nake, F.: Informatik und die Maschinisierung von Kopfarbeit, in: Coy u.a. 1992, S. 181-204

[Naur 1985]
Naur, P.: Programming as theory building. Microprocessing and Microprogramming, Amsterdam

[Neef 1982]
Neef, W.: Ingenieure. Entwicklung und Funktion einer Berufsgruppe, Köln

[Nielsen 1992]
Nielsen, J.: The usability engineering life cycle, in: Computer, März 1992, S. 12-22

[Norman 1986]
Norman, D.A.: Cognitive engineering, in: Norman/Draper 1986

[Norman/Draper 1986]
Norman, D.A., Draper, S.W. (Hrsg.): User centred system design, Hillsdale

[Novik 1993]
Novik, D.: Participatory conversation in PD, in: Communications of the ACM, June 1993, Vol. 36, No.4, S.93

[Nuissl/Siebert/Weinberg 1988]
Nuissl, E., Siebert, H., Weinberg, G. (Hrsg.): Literatur- und Forschungsreport Weiterbildung (Schwerpunktheft Weiterbildung)

[Nullmeier 1988]
Nullmeier, E.: Gestaltung rechnergestützer Arbeitsplätze im Büro, in: Nullmeier/Rödiger 1988, S. 109-121

[Nullmeier/Rödiger 1988]
Nullmeier, E., Rödiger, K.-H. (Hrsg.): Dialogsysteme in der Arbeitswelt, Mannheim

[Nygaard 1986]
Nygaard, K.: Program development as a social activity, in: Kugler 1986, S. 189-198

[Nygaard/Sørgaard 1989]
Nygaard, K., Sorgaard, P.: The perspective concept in informatics, in: Bjerknes/Ehn/Kyng 1989, S. 371-395

[Oberquelle 1991]
Oberquelle, H.: MCI - Quo Vadis? Perspektiven für die Gestaltung und Entwicklung der Mensch-Computer- Interaktion, in: Ackermann/Ulich 1991, S. 9-24

[Oberquelle 1991a]
Oberquelle, H.: Kooperative Arbeit und Computerunterstützung. Stand und Perspektiven, Göttingen, Stuttgart

[Oberquelle 1991b]
Oberquelle, H.: Perspektiven der Mensch-Computer-Interaktion und kooperative Arbeit, in: Frese u.a. 1991, S.45-56

[Oberquelle 1991c]
Oberquelle, H.: Kooperative Arbeit und menschengerechte Groupware als Herausforderung für die Software-Ergonomie, in: Oberquelle 1991a, S. 1-10

[Oberquelle 1991d]
Oberquelle, H.: CSCW- und Groupware-Kritik, in: Oberquelle 1991a, S. 37-62

[Oesterreich/Volpert 1991]
Oesterreich, R., Volpert, W.: VERA Version 2, Berlin

[Oppermann 1983]
Oppermann, R.: Forschungsgegenstand und Perspektiven partizipativer Systementwicklung, München

[Oppermann u.a. 1988]
Oppermann, R., Murchner, B., Paetau, M., Pieper, M., Simm, P., Stellmacher, I.: Evaluation von Dialogsystemen. Der software-ergonomische Leitfaden EVADIS II, Berlin

[Oppermann u.a. 1992]
Oppermann, R., Murchner, B., Reiterer, H., Koch, M.: Softwareergonomische Evaluation. Der Leitfaden EVADIS II, Berlin

[Oppermann/Thomas 1993]
Oppermann, R., Thomas, C.: Auf dem Weg zur benutzerspezifischen Systemanpassung, in: Coy u.a. 1993, S. 442-462

[Ortlieb/Holz auf der Heide 1993]
Ortlieb, S., Holz auf der Heide, B.: Benutzer bei der Software-Entwicklung angemessen beteiligen - Erfahrungen und Ergebnisse mit verschiedenen Konzepten, in: Rödiger 1993, S. 249-260

[Ortmann u.a. 1990]
Ortmann, G., Windeler, A., Becker, A., Schulz, H.-J.: Computer und Macht in Organisationen. Mikropolitische Analysen, Opladen

[Otte 1994]
Otte, M.: Das Formale, das Soziale und das Subjektive. Eine Einführung in die Philosophie und Didaktik der Mathematik, Frankfurt a.M.

[Owen 1986]
Owen, D.: Naive theories of computation, in: Norman/Draper 1986, S. 187-200

[Paetau 1990]
Peatau, M.: Mensch-Maschine Kommunikation. Software, Gestaltungspotentiale, Sozialverträglichkeit, Frankfurt a.M., New York

[Paetau 1991]
Paetau, M.: Mensch-Maschine Kommunikation im Spannungsfeld zwischen Individualisierung und Standartisierung. GMD Arbeitspapier 520, Sankt Augustin

[Paetau 1991a]
Paetau, M.: Systemanpassung als Kooperationsproblem, in: Ackermann/Ulich 1991, S. 281-290

[Paetau 1991b]
Paetau, M.: Kooperative Konfiguration - Ein Konzept zur Systemanpassung an die Dynamik kooperativer Arbeit, in: Friedrich/Rödiger 1991, S. 137-152

[Paetau 1991c]
Paetau, M.: Zur Relevanz individualisierbarer Software in neueren Gestaltungskonzeptionen der Büroarbeit, in: Frese u.a. 1991, S. 253-266

[Pasch 1989]
Pasch, J.: Mehr Selbstorganisation in Softwareentwicklungsprojekten, in: Softwaretechnik-Trends; Mitteilungen der Fachgruppe "Software-Engineering" der Gesellschaft für Informatik, September 1989

[Pehl 1989]
Pehl, K.: Schlüsselqualifikationen und Praxisorientierung, in: Meisel u.a. 1989

[Perin 1991]
Perin, P.: Electronic social fields in bureaucracies, in: Communications of the ACM, Dec.1991, Vol.34, No.12

[Pflüger/Schurz 1988]
Pflüger, J., Schurz, R.: Algorithmus und Ambivalenz, in: Krafft/Ortmann 1988

[Piaget 1976]
Piaget, J.: Die Äquilibration der kognitiven Strukturen, Stuttgart

[Piaget 1981]
Piaget, J.: Sprache und intellektuelle Operationen, in: Furth 1981

[Piaget 1981a]
Piaget, J.: Assimilation und Wahrnehmung, in: Furth 1981

[Piepenburg 1991]
Piepenburg, U.: Ein Konzept von Kooperation und seine Implikationen für die technische Unterstützung kooperativer Prozesse, in: Oberquelle 1991a, S. 79-98

[Piepenburg 1991a]
Piepenburg, U.: Ein Konzept von Kooperation und die technische Unterstützung kooperativer Prozesse im Bürobereich, in: Friedrich/Rödiger 1991, S. 79-94

[Preglau 1989]
Preglau, M.: Kritische Theorie: Jürgen Habermas, in: Morel u.a. 1989

[Projektgruppe Handlungslernen 1984]
Gerds, P., Rauner, F., Weisenbach, K.: Lernen durch Handeln in der beruflichen Bildung, Wetzlar

[Prümper 1993]
Prümper, J.: Benutzerorientierte, iterative Software-Entwicklung in der Praxis, in: Coy u.a. 1993, S.630-647

[Raasch 1991]
Raasch, J.: Systementwicklung mit Strukturierten Methoden. Ein Leitfaden für Praxis und Studium, Müchen, Wien

[Raeithel 1991]
Raeithel, A.: Zur Ethnographie der kooperativen Arbeit, in: Oberquelle 1991a, S. 99-112

[Raeithel 1992]
Reaithel, A.: Ein kulturhistorischer Blick auf rechnergestützte Arbeit, in: Coy u.a. 1992, S.125-140

[Rammert 1992a]
Rammert, W.: Neue Technologie - neue Begriffe? Lassen sich die Technologien der Informatik mit den traditionellen Konzepten der Arbeits- und Industriesoziologie noch angemessen erfassen?, in: Malsch/Mill 1992; S. 29-52

[Rauterberg 1991]
Rauterberg, M.: Benutzungsorientierte Benchmark-Tests: eine Methode zur Benutzerbeteiligung, in: Ackermann/Ulich 1991, S. 96-107

[Rauterberg 1992a]
Rauterberg, M.: Messung der Gebrauchstauglichkeit interaktiver Software, in: Görke/Rininsland/Syrbe 1992, S.211-221

[Rauterberg 1992b]
Rauterberg, M.: Läßt sich die Gebrauchstauglichkeit interaktiver Software messen? Und wenn ja, wie?, in: Ergonomie & Informatik, 16 (1992), S. 3-18

[Rauterberg 1992c]
Rauterberg, M.: Partizipative Modellbildung zur Optimierung der Softwareentwicklung, in: Studer 1992, S. 113-128

[Rauterberg 1993]
Rauterberg, M.: Anforderungen an die Prozeßgestaltung der Softwareentwicklung, in: Coy u.a. 1993, S. 592-605

[Rauterberg/Mollenhauer/Spinas 1993]
Rauterberg, M., Mollenhauer, R., Spinas, P.: Phasenmodell ist OUT. Benutzerbeteiligung jetzt auch bei Standardsoftware-Entwicklung, in: Coy u.a. 1993, 572-591

[Reetz 1989]
Reetz, L.: Zum Konzept der Schlüsselqualifikationen in der Berufsausbildung (Teil 1), in: Berufsbildung in Wissenschaft und Praxis 5/89

[Reetz/Reitmann 1990]
Reetz, L., Reitmann, T. (Hrsg.): Schlüsselqualifikationen, Hamburg

[Reisin 1992]
Reisin, F.- M.: Kooperative Gestaltung in partizipativen Softwareprojekten, Frankfurt a.M.

[Reisin 1992a]
Reisin, F.-M.: Anicipating reality construction, in: Floyd u.a. 1992; S. 312-225

[Reisin 1992b]
Reisin, F.-M.: Kooperation in Software-Projekten - Von der Betroffenenpartizipation zur partizipativen Betroffenheit, in: Trautwein-Kalms 1992

[Reisin 1993]
Reisin, F.M.: Software-Ergonomie braucht Partizipation, Manuskript

[Reisin 1994]
Reisin, F-M.: Software-Ergonomie braucht Partizipation, in: Eberle u.a. 1994

[Reuter 1990]
Reuter, A. (Hrsg.): Informatik auf dem Weg zum Anwender, Berlin

[Rock u.a. 1990]
Rock, R., Ulrich, P., Witt, F. (Hrsg.): Strukturwandel der Dienstleitungsrationalisierung, Frankfurt(M), New York

[Röder 1989]
Röder, S.: Personale Fähigkeiten in formalen Welten - "Schlüssel" zur informatisierten Wirklichkeit, in: Meisel u.a. 1989

[Rödiger 1988]
Rödiger, K.-H.: Das Arbeitsanalyseverfahren VERA/B in der Software-Entwicklung, in: Nullmeier/Rödiger 1988, S. 185-204

[Rödiger 1993]
Rödiger, K.H. (Hrsg.): Software-Ergonomie ´93. Von der Benutzungsoberfläche zur Arbeitsgestaltung, Stuttgart

[Rödiger u.a. 1991]
Rödiger, K.-H., Hampe-Neteler, W., Piepenburg, U.: Software-Ergonomie. Gestaltungsgrundsätze der DIN-Norm 66234, Teil 8, und ihre Umsetzung., in: Informationen zur Technologieberatung; 9.Technologieberatungsstelle beim DGB Landesbezirk NRW, Oberhausen

[Rödiger/Piepenburg 1989]
Rödiger, K.-H., Piepenburg, U.: Prüfung von Software auf die Grundsätze ergonomischer Dialoggestaltung, in: Maaß/Oberquelle 1989, S.163-173

[Röhner 1992]
Röhner, F.: Die Dynamik komplexer Systeme: Synergetik als Beschreibungsmethode, in: Langenheder/Müller/Schinzel 1992

[Rolf 1986]
Rolf, A. (Hrsg.): Neue Techniken alternativ, Hamburg

[Rolf 1992]
Rolf, A.: Sichtwechsel - Informatik als (gezähmte) Gestaltungswissenschaft, in: Coy u.a. 1992, S.17-32

[Rolf 1992a]
Rolf, A.: Informatik als Gestaltungswissenschaft, in: Langenheder/Müller/Schinzel 1992

[Rolf 1993]
Rolf, A.: Informatik und Gestaltung, in: InfoTech (Jg. 5) Heft 4, Dez. 1993- Febr. 1994, S. 16-22

[Rolf u.a. 1991]
Rolf, A., Berger, P., Klischewski, R., Kühn, M., Massen, A., Winter, R.: Technikleitbilder und Büroarbeit. Zwischen Werkzeugperspektive und globalen Vernetzungen

[Ruben/Wolter 1969]
Ruben, P., Wolter, H.: Modell, Modellmethoden und Wirklichkeit, in: Deutsche Zeitschrift für Philosophie 1969, S. 1225-1239

[Rumpf 1986]
Rumpf, H.: Mit fremdem Blick. Stücke gegen die Verbiederung der Welt, Weinheim, Basel

[Sabel/Piore 1985]
Sabel, C., Piore, J.: Das Ende der Massenproduktion, Berlin

[Sandkühler, H.J. 1990]
Sandkühler, H.-J.: Europäische Enzyklopädie zu Philosophie und Wissenschaft, Band 2, Hamburg

[Schade 1993]
Schade, G.: Konzept für sozialorientierte Informatikausbildung an der Technischen Universität Ilmenau - Thesen, in: InfoTech (Jg. 5) Heft 4, Dez. 1993 -Febr. 1994

[Scheer 1987/1988]
Scheer, A.-W.: CIM. Der computergesteuerte Industriebetrieb, Berlin, Heidelberg, New York

[Scheerer 1993]
Scheerer, E.: Mentale Repräsentationen in interdisziplinärer Perspektive, in: Zeitschrift für Psychologie 2/93

[Schimank 1981]
Schimank, U.: Identitätsbehauptungen in der Arbeitsorganisation - Individualität in der Formalstruktur, Frankfurt a.M., New York

[Schimank 1985]
Schimank, U.: Funktionale Differenzierung und reflexiver Subjektivismus. Zum Entsprechungsverhältnis von Gesellschafts- und Identitätsform, in: Soziale Welt, 36 (1985), S. 447-465

[Schimank 1986]
Schimank, U.: Technik, Subjektivität und Kontrolle in formalen Organisationen - eine Theorieperspektive, in: Seltz/Mill/Hildebrandt 1986

[Schimank 1988]
Schimank, U.: Gesellschatliche Teilsysteme als Akteurfiktionen, in: Kölner Zeitschrift für Soziologie und Sozialpsychologie; No3, Vol 40, 1988

[Schindler/Belke 1991]
Schindler, R., Belke, S.: Lerngewinn aus Beispielen in Abhängigkeit von der Lernform und Aufmerksamkeitslenkung, in: Ackermann/Ulich 1991, S. 322-331

[Schinzel 1993]
Schinzel, B.: Zur Gleichstellung von Frauen und Männern in der Informatik, in: InfoTech (Jg. 5), Heft 4, Dez. 1993 - Febr. 1994, S. 7-15

[Schlicksupp 1983]
Schlicksupp, H.: Innovation, Kreativität und Ideenfindung, 3. Aufl., Würzburg

[Schmidt 1986]
Schmidt, G.: Einverständnishandeln - ein Konzept zur ´handlungsnahen´ Untersuchung betrieblicher Entscheidungsprozesse, in: Seltz/Mill/Hildebrandt 1986, S. 57-68

[Schmidt/Rasmussen 1991]
Schmidt, V., Rasmussen, J.: Unraveling work organizations, in: Frese u.a. 1991, S. 213-232

[Schmiede 1988]
Schmiede, R.: Arbeit und Subjektivität, o.O.

[Schmiede 1992]
Schmiede, R.: Information und kapitalistische Produktionsweise. Entstehung der Informationstechnik und Wandel der gesellschaftlichen Arbeit, in: Malsch/Mill 1992; S. 53-86

[Schubert 1989]
Schubert, R.: Zur Möglichkeit von Technikphilosophie. Versuch einer modernen Kritik der Urteilskraft, Wien

[Schulz-Zander u.a. 1993]
Schulz-Zander, R. u. a.: Veränderte Sichtweisen für den Informatikunterricht. GI-Empfehlungen für das Fach Informatik in der Sekundarstufe II allgemeinbildender Schulen, in: Informatik-Spektrum (1993) 16, S. 349-356

[Schurer 1986]
Schurer, B.: Grundlagen einer subjektorientierten Didaktik beruflichen Lernens und Lehrens in Unterricht und Unterweisung, in: Zeitschrift für Berufs- und Wirtschaftspädagogik, 82.Jg., H.8, S. 675-690

[Schwill 1993]
Schwill, A.: Fundamentale Ideen der Informatik, in: Zentralblatt für Didaktik der Mathematik, H. 93/1, S. 20-31

[Schwitalla u.a. 1990]
Schwitalla, U., Wicke, W., Jansen, K.-D.: Entwicklung computergestützter Arbeitssysteme unter der Beteiligung der Betroffenen, Opladen

[Seetzen 1992]
Seetzen, J.: Information, Kommunikation, Organisation - Anmerkungen zur "Theorie der Informatik", in: Coy u.a. 1992, S. 83-96

[Seetzen 1993]
Seetzen, J.: Komplexitätsbewältigung als Theorie und Praxis der Informatik und ihre Konsequenzen für die Ausbildung von Informatikern, in: InfoTech; Jg.5, Heft 4, Dez. 1993 - Febr. 1994, S.5-6

[Seitz 1993]
Seitz, D.: "Per Order de Mufti läuft nichts". Zur sozialen Steuerung betrieblicher Gestaltungsprozesse, Berlin

[Sell 1988]
Sell, R.: Angewandtes Problemlöseverhalten, Frankfurt a.M., New York

[Sell/Fuchs-Frohnhofen 1993]
Sell, R., Fuchs-Frohnhofen, P.: Gestaltung von Arbeit und Technik durch Beteiligungsqualifizierung, Opladen

[Sell/Henning 1993]
Sell, R., Henning, K.: Lernen und Fertigen. Konzepte zur Facharbeitergerechten Gestaltung von Technik, Organisation und Qualifizierung am Beispiel rechnergestützter Fertigung, Aachen

[Seltz/Mill/Hildebrandt 1986]
Seltz, R., Mill, U., Hildebrandt, E.: Organisation als soziales System, Berlin

[Semmer/Pfäfflin 1979]
Semmer, N., Pfäfflin, M.: Interaktionstraining. Ein handlungstheoretischer Ansatz zum Training sozialer Fertigkeiten, Weinheim, Basel

[Senghaas-Knobloch/Volmberg 1990]
Senghaas-Knobloch, E., Volmberg, B.: Technischer Fortschritt und Verantwortungsbewußtsein. Die gesellschaftliche Verantwortung von Ingenieuren, Opladen

[Serres 1987]
Serres, M.: Der Parasit, Frankfurt a.M.

[Shneiderman 1983]
Shneiderman, B.: Direct Manipulation: A step beyond programming languages, in: IEEE Computer 16 (8), S. 57-69 (1983)

[Shneidermann 1992]
Shneidermann, B.: Designing the user interface: Strategies for effective human-computer interaction, Reading, Massachusetts

[Siebert 1985]
Siebert, H.: Identitätslernen in der Diskussion, Frankfurt a.M.

[Siefkes 1992]
Siefkes, D.: Formalisieren und Beweisen. Logik für Informatiker. 2.,verb. Auflage

[Siefkes 1993]
Siefkes, D.: Formale Methoden und kleine Systeme. Lernen, leben und arbeiten in formalen Umgebungen, Wiesbaden

[Siefkes 1993a]
Siefkes, D.: Evolutionäre Denkmodelle in der Informatikerausbildung, in: InfoTech; Jg. 5 Heft 4 Dez.93-Febr. 94, S. 26-33

[Simon 1981]
Simon, D.: Hackers Arbeitspsychologie - ein theoretisches Konzept für humane Arbeitsgestaltung?, in: Zeitschrift für Berufs- und Wirtschaftspädagogik, H.7 (1981), S. 504-519

[Simonsmeier 1992]
Simonsmeier, W.: Arbeitszufreidenheit und Überforderung der Software-Entwicklerinnen und Entwickler, in: Trautwein-Kalms 1992

[Skarpelis 1993]
Skarpelis, C.: Aufgabengerechte Software: Bestand, Bedarf, in: Coy u.a. 1993, S. 44-88

[Sledge 1992]
Sledge, C. (Hrsg.): Software Engineering Education. SEI Conference 1992, San Diego, California, USA, Oct. 1992, Proceedings, edited by Sledge, C., in: Lecture Notes in Computer Science 640, Berlin, Heidelberg, New York

[Söltenfuß 1983]
Söltenfuß, G.: Grundlagen handlungsorientierten Lernens, Bad Heilbrunn

[Spillner 1994]
Spillner, A.: Kann eine Krise 25 Jahre dauern?, in: Informatik-Spektrum (1994) 17, S. 48-52

[Spinas u.a. 1993]
Spinas, P., Rauterberg, M., Strohm, O., Waeber, D., Ulich, E.: Benutzerorientierte Software-Entwicklung. Konzepte, Methoden und Vorgehen zur Benutzerbeteiligung, in: Schriftenreihe Mensch,Technik, Organisation (Hrsg. E. Ulich); Band 3, Zürich, Stuttgart

[Spinas/Waeber 1991]
Spinas, P., Waeber, D.: Benutzerbeteiligung aus der Sicht von Endbenutzern, Software-Entwicklern und Führungskräften, in: Ackermann/Ulich 1991, S. 36-45

[Spitta 1989]
Spitta, Th.: Software-Engineering und Prototyping, Berlin, Heidelberg, New York

[Strübing 1988]
Strübing, J.: Programmieren in einer betrieblichen Sonderkultur? Überlegungen zu Arbeitsstil und Fachkultur in der Programmierarbeit, in: Innovation, Subjektivität und Verantwortung. Tagungsbericht, Universität Kassel

[Strübing 1992]
Strübing, J.: Arbeitsstil und Habitus. Zur Bedeutung kultureller Phänomene in der Programmierarbeit, Kassel

[Strübing 1993]
Strübing, J.: Subjektive Leistungen im Arbeitsprozeß - dargestellt anhand einer empirischen Untersuchung von Arbeitsstilen in der Programmierarbeit, Opladen

[Sydow 1985]
Sydow, J.: Der soziotechnische Ansatz der Arbeits- und Organisationsgestaltung, Frankfurt

[Tenorth 1992]
Tenorth, H.-E.: Intention - Funktion - Zwischenreich, in: Luhmann/Schorr 1992

[Tesch-Römer 1990]
Tesch-Römer, C.: Identitätsprobleme und Identitätsformation im mittleren Erwachsenenalter, Berlin

[Thomas/Elstermann 1986]
Thomas, H., Elstermann, G. (Hrsg.): Bildung und Beruf. Soziale und ökonomische Aspekte, Berlin

[Tillmann 1989]
Tillmann, K.-J.: Sozialisationstheorien, Hamburg

[Trautwein-Kalms 1992]
Trautwein-Kalms, G. (Hrsg.): KontrastProgramm Mensch-Maschine. Arbeiten in der HighTech-Welt, Köln

[Troll 1982]
Troll, L.: Arbeitsplatz Büro. Berufe, Qualifikationen und Arbeitsplatzsituation im Wandel, in: Mitteilungen zur Arbeitsmarkt- und Berufsforschung, H. 4/1982

[Twiehaus/Dostal 1990]
Twiehaus, H.-J., Dostal, W.: Computerberufe. Berufe und Bildung in der Datenverarbeitung, 2. Auflage, München

[Ulich 1988]
Ulich, E.: Arbeits- und organisationspsychologische Aspekte, in: Balzert u.a. 1988, S. 49-66

[Ulich 1989]
Ulich, E.: Arbeitspsychologische Konzepte der Aufgabengestaltung, in: Maaß/Oberquelle 1989, S. 51-68

[Ulich 1991]
Ulich, E.: Arbeitspsychologie, 2. Aufl., Stuttgart

[Ulich 1991a]
Ulich, E.: Gruppenarbeit - arbeitspsychologische Konzepte und Beispiele, in: Friedrich/Rödiger 1991, S. 57-78

[Ulich 1993]
Ulich, E.: Von der Benutzungsoberfläche zur Arbeitsgestaltung, in: Rödiger 1993, S. 19-30

[Urban 1993]
Urban, K.K.: Kreativitätsforschung, in: Psychologie in Erziehung und Unterricht, 40.Jg (1993), H.3

[Varela 1990]
Varela, F.J.: Kognitonswissenschaft - Kognitionstechnik, Frankfurt

[Vaske, H. 1994]
Vaske, H.: EU-Richtlinie soll Rechte der Endanwender absichern. Vorschrift zur Bildschirmarbeit birgt viel Zündstoff, in: Computerwoche 3, 21. Jg. 1994

[VDI 5005]
Verein Deutscher Ingenieure: Bürokommunikation. Software-Ergonomie in der Bürokommunikation, in: VDI-Richtlinien Nov. 1988

[VDI Nachrichten 46/1990]
Software-Projekte scheitern oft an mangelnder Abstimmung, in: VDI Nachrichten Nr. 46 vom 16. November 1990, S. 34

[Viereck 1993]
Viereck, A.: Design von Benutzungsoberflächen als ingenieurmäßiger Prozeß, in: Rödiger 1993, S. 311-320

[Volmberg/Senghaas-Knobloch/Leithäuser 1988]
Volmberg, B., Senghaas-Knobloch, E., Leithäuser, Th.: Technischer Fortschritt und Verantwortungsbewußtsein, Bremen

[Volpert 1974]
Volpert, W.: Handlungsstrukturanalyse als Beitrag zur Qualifikationsforschung, Köln

[Volpert 1983]
Volpert, W.: An den Grenzen des Modells der hierarchisch-sequentiellen Handlungsorganisation, in: Berliner Hefte zur Arbeits- und Sozialpsychologie 1983 Nr.3, S. 1-29

[Volpert 1985]
Volpert, W.: Zauberlehrlinge. Die gefährliche Liebe zum Computer, Weinheim

[Volpert 1992]
Volpert, W.: Erhalten und gestalten - von der notwendigen Zähmung des Gestaltungsdrangs, in: Coy u.a. 1992, S. 171-180

[Volpert 1992a]
Volpert, W.: Die Kontrastive Aufgabenanalyse im Kontext der Diskussion zwischen Arbeitspsychologen und Informatikern, in: Malsch/Mill 1992, S. 185-195

[Volpert 1992b]
Volpert, W.: Wie wir handeln - was wir können, Heidelberg

[Volpert 1993]
Volpert, W.: Arbeitsinformatik- der Kooperationsbereich von Informatik und Arbeitswissenschaft, in: Zeitschrift für Arbeitswissenschaft 2/1993

[Voyat 1982]
Voyat, G.: Entwicklung in der kognitiven und in der sozialen Dimension. Eine neue Perspektive, in: Edelstein/Keller 1982, S.219-236

[Wagner 1991]
Wagner, I.: Groupware zur Entscheidungsunterstützung als Element von Organisationskultur, in: Kooperatives Arbeiten und Computerunterstützung. Stand und Perspektiven, Göttingen

[Wagner 1992]
Wagner, I.: Formalisierte Kooperation. Organisationskulturelle Aspekte der Computerunterstützung kommunikativer Verständigungsprozesse, in: Malsch/Mill 1992; S. 197-218

[Wagner 1993]
Wagner, I.: A web of Fuzzy problems: Confronting the ethical issues, in: Communications of the ACM, Vol. 36, No.4, June 1993, S. 94-101

[Waldvogel 1990]
Waldvogel, M.: Das Einzigartige und die Sprache. Ein Essay, Wien

[Watzlawick u.a. 1972]
Watzlawick, P., Beavin, J.H., Jackson, D.D.: Menschliche Kommunikation. Formen, Störungen, Paradoxien, Bern, Stuttgart, Wien

[Wehrsig/Tacke 1992]
Wehrsig, C., Tacke, V.: Funktionen und Folgen informatisierter Organisationen, in: Malsch/Mill 1992; S. 219-241

[Weisbecker 1993]
Weisbecker, A.: Unterstützungswerkzeuge zur benutzergerechten Gestaltung der Mensch-Computer-Schnittstelle, in: Coy u.a. 1993, S.184-199

[Weisbecker 1993a]
Weisbecker, A.: Integration von software-ergonomischem Wissen in die Systementwicklung, in: Rödiger 1993, S. 299-310

[Weisenbach/Rauner/Gerds 1988]
Weisenbach, K., Rauner, F., Gerds, P.: Handlungsorientierter Fachunterricht in Kraftfahrzeug-Mechanikerklassen, Bremen

[Weizenbaum 1977]
Weizenbaum, J.: Die Macht der Computer und die Ohnmacht der Vernunft, Frankfurt a.M.

[Weltz/Lullies/Ortmann 1991]
Weltz, F., Lullies, V., Ortmann, R.G.: Software-Entwicklung als Prozeß der Arbeitsstrukturierung, in: Ackermann/Ulich 1991

[Weltz/Ortmann 1992]
Weltz, F., Ortmann, R.G.: Das Softwareprojekt. Projektmanagment in der Praxis, Frankfurt a.M.

[Weltz/Bullinger]
Weltz, F, Bullinger, H.J.: Nutzerbeteiligung oder kooperative Systementwicklung?, in: Office Management H.3/1990

[Wendt 1993]
Wendt, S.: Defizite im Software Engineering, in: Informatik Spektrum, Februar 1993, S. 34-38

[Wicke 1986]
Wicke, W.: Partizipative Sytemgestaltung: Rationalisierung oder Humanisierung?, in: Rolf 1986, S.134-145

[Wiener 1948]
Wiener, N.: Cybernetics, New York

[Wiesner 1992]
Wiesner, B.: Evolutionäre Expertensystementwicklung und Softwarequalität, in: Langenheder/Müller/Schinzel 1992, S. 253-257

[Williams/Begg 1993]
Williams, M.G., Begg, V.: Translation between software designers and users, in: Communications of the ACM, Vol. 36, No. 4, June 1993, S. 102-103

[Windolf 1981]
Windolf, P.: Berufliche Sozialisation. Zur Produktion des beruflichen Habitus, Stuttgart

[Winograd/Flores 1989]
Winograd, T., Flores, F.: Erkenntnis Maschinen Verstehen, Berlin

[Womack u.a. 1991]
Womack, J.P., Jones, D.T., Roos, D.: Die zweite Revolution in der Autoindustrie: Konsequenzen aus der weltweiten Studie aus dem Massachussetts Institute of Techology, Frankfurt a.M., New York

[Wood 1986]
Wood, S.: Neue Technologien, Arbeitsorganisation und Qualifikation: Die britische Labour-Process-Debatte, in: PROKLA 62

[Zabeck 1989]
Zabeck, J.: "Schlüsselqualifikationen" - Zur Kritik einer didaktischen Zielformel, in: Wirtschaft und Erziehung, 41. Jg. (1989), H.3.; S. 77-89

[Zemanek 1986]
Zemanek, H.: Information und Ingenieurswissenschaft, in: Folberth/Hackl 1986, S.17-52

[Zendler 1993]
Zendler, A.: Benutzerorientierte Softwaregestaltung und Client/Server-Anwendungen mit MAESTRO II, in: Coy u.a. 1993

[Ziegler 1993]
Ziegler, J.: Benutzergerechte Softwaregestaltung im VDI-Gemeinschaftsausschuß Bürokomunikation - VDI 5005 "Software-Ergonomie in der Bürokommunikation", in: Coy u.a. 1993, S. 140-151

[Ziegler/Koller 1993]
Ziegler, J., Koller, F.: Wissensorientierte Unterstützung von Arbeit und Lernen - Technologien und Einsatzkriterien, in: Coy u.a. 1993, S. 369-413

[Zölch 1992]
Zölch, M.: Partizipation contra krisengeleitete Aufgabenbewertung - eine Scheinalternative?, in: Langenheder/Müller/Schinzel 1992, S. 243-247

[Zölch/Dunckel 1991]
Zölch, M., Dunckel, H.: Kontrastive Aufgabenanalyse - Ergebnisse des Verfahrenseinsatzes - Praxisrelevanz, in: Ackermann/Ulich 1991, S.363-372

[Züllighoven 1992]
Züllighoven, H.: Umgang mit Software oder: Software als Werkzeug und Material, in: Coy u.a. 1992, S. 141-156

[Zündorf 1986]
Zündorf, L.: Macht, Einfluß, Vertrauen und Verständigung. Zum Problem der Handlungskoordinierung in Arbeitsorganisationen, in: Seltz/Mill/Hildebrandt 1986