

Softwareentwicklungsverträge

Inaugural-Dissertation zur Erlangung des Doktorgrades am Fachbereich
Rechtswissenschaft der Justus-Liebig-Universität Gießen

vorgelegt von Peter Kath

Gießen 2006

Guruji Dr. Shri Balaji Tambe und meiner Familie

Inhaltsübersicht

Kapitel 1: Gang der Untersuchung, Bedeutung der IT-Branche und Einführung in die Begriffe Software und Leistungsbestimmung.....	3
Kapitel 2: Softwareentwicklung in der Praxis.....	10
Kapitel 3: Rechtliche Rahmenbedingungen für Softwareentwicklungsverträge.....	36
Kapitel 4: Empirischer Befund in den USA.....	59
Kapitel 5: Rechtliche Rahmenbedingungen in Deutschland.....	99
Kapitel 6: Empirischer Befund.....	130
Kapitel 7: Vergleichende Darstellung – Methoden der Leistungsbestimmung.....	192
Kapitel 8: Schlussfolgerungen.....	212
Kapitel 9: Zusammenfassende Würdigung.....	223
Literaturverzeichnis.....	237

Inhaltsverzeichnis

Kapitel 1: Gang der Untersuchung, Bedeutung der IT-Branche und Einführung in die Begriffe Software und Leistungsbestimmung.....	3
A. Gang der Untersuchung.....	3
B. Bedeutung der Informationstechnologie.....	3
C. Erscheinungsformen von Software.....	5
D. Die Bedeutung der Leistungsbestimmung.....	8
Kapitel 2: Softwareentwicklung in der Praxis.....	10
A. Theorie des Softwareengineering.....	10
I. Einführung.....	10
II. Modelle für die Erstellung von Software.....	12
III. Für den Untersuchungsgegenstand besonders bedeutsame Aspekte der Softwareentwicklung.....	20
1. Der lange Weg zum Pflichtenheft – Requirements Engineering.....	20
a) Einführung.....	20
b) Aufwand.....	22
c) Zeitpunkt und Zuständigkeit.....	23
d) Voraussetzung für die Erstellung.....	24
e) Anforderungen und Abnahme.....	25
2. Mitarbeit des Softwarebestellers.....	26
3. Änderungen.....	27
a) Psychologie des Entwicklungsprozesses.....	28
b) Kritische Faktoren in der Softwareentwicklung.....	30
c) Zusammenfassung und Ergebnis.....	32
Kapitel 3: Rechtliche Rahmenbedingungen für Softwareentwicklungsverträge.....	36
A. Einordnung und Bedeutung der Rechtsquellen.....	36
B. Performance.....	38
C. Statute Law.....	40
I. U.C.C.....	40
1. Anwendbarkeit.....	40
2. Anwendbarkeit des U.C.C. auf Software.....	41
3. Anwendbarkeit auf Softwareentwicklung.....	43

4. Regeln zur Leistungsbestimmung.....	45
II. Convention on Contracts for the International Sale of Goods (CISG)	47
III. Uniform Computer Information Transaction Act (UCITA).....	49
D. Common Law.....	49
I. Restatements.....	49
II. Case Law.....	53
E. Zusammenfassung und Ergebnis.....	56
Kapitel 4: Empirischer Befund in den USA.....	59
A. Musterentwürfe.....	59
I. Vertragsart.....	60
II. Bestandteile der Lieferung	62
III. Spezifikationen.....	65
IV. Besondere Pflichten des Softwarehauses.....	68
V. Besondere Pflichten des Softwarebestellers.....	72
VI. Veränderungen (Change Management).....	74
VII. Zusammenhang zwischen Leistungsbestimmung und Einstandspflicht.....	77
B. Beispiele aus der Praxis.....	79
I. Vertragsart.....	80
II. Bestandteile der Lieferung	83
III. Spezifikationen.....	84
IV. Besondere Pflichten des Softwarehauses.....	87
V. Besondere Pflichten des Softwarebestellers.....	91
VI. Veränderungen (Change Management).....	92
VII. Zusammenhang zwischen Leistungsbestimmung und Einstandspflicht.....	93
C. Zusammenfassung und Ergebnis.....	96
Kapitel 5: Rechtliche Rahmenbedingungen in Deutschland.....	99
A. Einordnung und Bedeutung der Rechtsquellen.....	99
I. Rechtsprechung und Gesetz	99
II. Das Prinzip Privatautonomie.....	100
III. Auslegungsgrundsätze.....	101
B. Anwendbares Recht.....	104

I. Einordnung des Vertragstypus.....	104
1. Notwendigkeit.....	104
2. Typenauswahl und Einordnungskriterien.....	105
b) Was ist Software.....	108
c) Zweck der Softwareentwicklung.....	109
d) Urheberrechtliche Einflüsse	110
3. Ergebnis.....	111
II. Der Mangelbegriff bei der Softwareentwicklung.....	111
III. Leistungsbestimmung.....	112
1. Bedeutung insbesondere bei der Softwarestellung	112
2. Zeitpunkt und Anforderungen.....	112
3. Gesetzliche Regeln nach modifiziertem Kaufvertragsrecht (§ 651 Abs.1 S.3 BGB) und Werkvertragsrecht (§§ 631ff BGB)	113
4. In sonstigen Konstellationen („Lizenzvertrag“, §§ 705ff, F&E- Vertrag, „dynamischer Vertrag“).....	116
IV. Leistungsbestimmung durch das „Pflichtenheft“	117
1. Begriff und Bedeutung	117
2. Verantwortungszuweisung.....	119
V. Beratungs- und Mitwirkungspflichten.....	120
1. Beratungs- und Dokumentationspflichten des Auftragnehmers	121
2. Mitwirkungspflichten des Auftraggebers	122
VI. Änderungen der Leistungsbestimmung.....	123
VII. UN-Kaufrecht (CISG).....	124
C. Zusammenfassung und Ergebnis.....	126
Kapitel 6: Empirischer Befund.....	130
A. Musterentwürfe.....	130
I. Vertragsart.....	131
II. Bestandteile der Lieferung	136
III. Spezifikationen.....	139
IV. Besondere Pflichten des Softwarehauses.....	145
V. Besondere Pflichten des Softwarebestellers.....	152
VI. Veränderungen (Change Management).....	159

VII. Zusammenhang zwischen Leistungsbestimmung und Einstandspflicht	167
B. Beispiele aus der Praxis.....	169
I. Vertragsart.....	170
II. Bestandteile der Lieferung	174
III. Spezifikationen und Leistungsmaßstab.....	178
IV. Besondere Pflichten des Softwarehauses.....	180
V. Besondere Pflichten des Softwarebestellers.....	182
VI. Veränderungen (Change Management).....	185
VII. Sonstiges.....	187
C. Zusammenfassung und Ergebnis.....	188
Kapitel 7: Vergleichende Darstellung – Methoden der Leistungsbestimmung.....	192
A. Zielpräzision.....	194
B. Reaktionsmechanismen bei Änderungen.....	198
C. Einbindung des Softwarebestellers - Mitwirkungspflichten.....	202
D. Qualitätskriterien.....	206
E. Vergütung	211
Kapitel 8: Schlussfolgerungen.....	212
A. Strukturelle Unterschiede – Spielräume und ihre Verwendung.....	212
I. Software ist keine Sache.....	212
II. Lösungen im Begriffssystem und Erfahrungssystem.....	215
III. Grenzen der Begriffssysteme	218
IV. Erfolgsorientierte Fehlerbestimmung.....	220
Kapitel 9: Zusammenfassende Würdigung.....	223
A. Gesamtbefund.....	223
I. Technik und Recht – Social Engineering und Software Engineering	223
II. Die Lösung des Konfliktes zwischen Planung und Wirklichkeit .	225
III. Interessenorientierte oder projektorientierte Vertragsgestaltung	227
B. Ergebnis	229
Literaturverzeichnis.....	237

Abkürzungen

A

a.A.	anderer Ansicht
a.a.O.	am angegebenen Ort
ABGB	Das Allgemeine bürgerliche Gesetzbuch (Österreich)
Abs.	Absatz
AcP	Archiv für die civilistische Praxis (Zeitschrift)
a.E.	am Ende
AG	Aktiengesellschaft
	Auftraggeber
AGB	Allgemeine Geschäftsbedingung(en)
AiB	Arbeitsrecht im Betrieb
ALI	American Law Institute
AN	Auftragnehmer
Anm.	Anmerkung
AnwK	Anwaltskommentar
Art.	Artikel
Aufl.	Auflage
Az	Aktenzeichen

B

BauR	Baurecht (Zeitschrift)
BB	Betriebsberater (Zeitschrift)
Bd.	Band
BGB	Bürgerliches Gesetzbuch
BGH	Bundesgerichtshof
BGHZ	Entscheidungen des Bundesgerichtshofs in Zivilsachen, amtliche Sammlung
BMI	Bundesministerium des Inneren
BVB	Besondere Vertragsbedingungen
BVerfG	Bundesverfassungsgericht
BvR	Aktenzeichen des Bundesverfassungsgerichts für Verfassungsbeschwerden
bzgl.	bezüglich
bzw.	beziehungsweise

C

CAD	Computer-Added Design
CISG	Convention on the International Sale of Goods
Corp.	Corporation
CORI	Contracting and Organizations Research Institute
CR	Computer und Recht (Zeitschrift)
Crt.	Court

D

ders.	derselbe
DFÜ	Datenfernübertragung
d.h.	das heißt
DIN	Norm des Deutschen Instituts für Normung e.V.
Diss.	Dissertation
Dist.	District
Distr.	District
DV	Datenverarbeitung

E

ECR	Entscheidungssammlung zum Computerrecht
-----	---

ed.	edition
EDGAR	US Securities and Exchange Commission
EDV	elektronische Datenverarbeitung
e.g.	exempli gratia – lat. (for example)
Einl.	Einleitung
ES	Erstellungsschein
F	
ff	und folgende
Fn	Fußnote
FS	Festschrift
G	
gem.	gemäß
GG	Grundgesetz
ggf.	gegebenenfalls
GWB	Gesetz gegen Wettbewerbsbeschränkungen
H	
HGB	Handelsgesetzbuch
Hrsg.	Herausgeber
h.M.	herrschende Meinung
I	
i.d.R.	in der Regel
IEEE	Institute of Electrical and Electronic Engineers
IHR	Internationales Handelsrecht (Zeitschrift)
Inc.	Incorporated
Incl.	einschließlich
i.S.	im Sinne
ISO	International Standard Organisation
IT	Informationstechnologie
ITIL	IT Infrastructure Library
ITRB	IT-Rechtsberater (Zeitschrift)
K	
K&R	Kommunikation und Recht (Zeitschrift)
KG	Kammergericht
KM	Konfigurationsmanagement
L	
L.	Law
LG	Landesgericht
ltd.	limited
M	
MAS	Member Administrative System
m.a.W.	mit anderen Worten
m.E.	meines Erachtens
MMR	Multimedia und Recht (Zeitschrift)
Mot.	Motive
MPEG	Moving Picture Experts Group (Standard zur Komprimierung von Dateien)
MüKo	Münchener Kommentar
m.w.Nachw.	mit weiteren Nachweisen
N	
NCCUSL	National Conference of Commissioners on Uniform State Laws
NJW	Neue Juristische Wochenschrift (Zeitschrift)

NJW-RR Nr.	Neue Juristische Wochenschrift-Rechtsprechungsreport (Zeitschrift) Nummer
O	
o.ä.	oder ähnlich
OLG	Oberlandesgericht
OLGR	Oberlandesgericht-Report (Entscheidungssammlung)
ORACLE	ein Datenbanksystem
P	
PC	Personal Computer
PM	Produktmanagement
Q	
QS	Qualitätssicherung
R	
Rev.	Review
RFP	Request for Proposal
Rn.	Randnummer
Rs.	Restatement
Rspr.	Rechtsprechung
S	
S.	Seite (Satz)
s.a.	siehe auch
SMS	Subscription Management Systems
s.o.	siehe oben
sog.	sogenannte(r)
s.u.	siehe unten
SWE	Softwareentwicklung
SRS	Software Requirement Specification
SWE	Softwareentwicklung
T	
Tb.	Tabelle
U	
u.a.	unter anderem
u.ä.	und ähnlich
U.C.C.	Uniform Commercial Code
UN	United Nations
US	United States of America
u.U.	unter Umständen
U.v.	Urteil vom
UCITA	Uniform Computer Information Transactions Act
UNCITRAL	United Nations Commission on International Trade Law
UrhG	Urhebergesetz
usw.	und so weiter
V	
v.	versus
v.a.	vor allem
VB	Vertragsbedingung
vgl.	vergleiche
W	
WAV	Format zur digitalen Speicherung von Audiodaten
WL	WestLaw (Juristische Datenbank)
WM	Wertpapier-Mitteilungen (Zeitschrift)

Z

z.B.	zum Beispiel
ZKM	Zeitschrift für Konflikt-Management
ZIP	Zeitschrift für das Wirtschaftsrecht und Insolvenzpraxis (Zeitschrift)

Kürzel für Muster und Verträge

AE	Softwareentwicklungsprojekt zwischen einem Telekommunikationsunternehmen in New Jersey (American Telephone and Telegraph Company) und einem Softwarehaus in Colorado (Evolving Systems Inc.)
AS	Musterentwurf von Amy Slater
BA	Musterentwurf von Bartsch
BVB	Besondere Vertragsbedingungen (BVB) - Erstellung
DaS	Musterentwurf von David F. Simon
DaS-U	Musterentwurf von David F. Simon („ <i>User Oriented</i> “)
DaS-V	Musterentwurf von David F. Simon („ <i>Vendor Oriented</i> “)
DS	Musterentwurf von Diane W. Savage
FDR I	First Data Resources Incorporation
FL	Musterentwurf eines in der Datenbank nicht genannten Autors (FL für Florida)
FP	Softwareentwicklungsprojekt zwischen einem Softwarehaus aus Nebraska (First Data Resources) und einem Softwarebesteller aus New York (Prodigy Services Company)
GL	Musterentwurf von Glöckler in: Sobola/Dobmeier
HE	Musterentwurf von Heppner
KO	Musterentwurf von Koch
Ma	Musterentwurf von Marly
MiN	Softwareentwicklungsprojekt zwischen einem Softwarehaus aus Mittelhessen und einem Produktionsbetrieb in Nordrhein-Westfalen
MiO	Softwareentwicklungsprojekt zwischen einem Softwarehaus aus Mittelhessen und einem Produktionsbetrieb in Österreich
OD	Softwareentwicklungsprojekt zwischen einem Softwarehaus in Manitoba/Canada (OMT Technologies Inc.) und einem Unternehmen in Los Angeles/California (DMX Inc.)
RB	Musterentwurf von Raysman/Brown
RB-U	Musterentwurf von Raysman/Brown („ <i>User Oriented</i> “)
RB-V	Musterentwurf von Raysman/Brown („ <i>Vendor Oriented</i> “)
SN	Musterentwurf von Schneider
SN-B	Musterentwurf von Schneider (am Besteller orientierte Variante)
SN-S	Musterentwurf von Schneider (am Softwarehaus orientierte Variante)
SÖ	Musterentwurf von Schröder
SüS	Softwareentwicklungsprojekt zwischen einem Softwarehaus in Südhessen und einem Softwarebesteller der öffentlichen Hand im Siegerland

Einleitung

Das Produkt Software existierte vor fünfzig Jahren nicht. Heute ist es aus dem Alltag nicht mehr wegzudenken, an seiner Entstehung sind Millionen von Menschen beteiligt und der ganz überwiegende Teil der Menschheit ist auf seine Existenz angewiesen, sei es zur Energieversorgung, zur Kommunikation oder zum Transport. Moderne Kraftwerke, Telekommunikationseinrichtungen und Fahrzeuge kommen ohne Software nicht aus. Noch deutlicher macht sich der Einfluss von Software dort bemerkbar, wo sie als Gebrauchsgegenstand zur Unterstützung oder als zentrales Medium für die eigene Arbeit dient. Neben der Hardware und den eigentlichen Daten nimmt die Software eine herausgehobene Position ein. Sie dient als Schnittstelle zwischen Mensch und Maschine und erfüllt im wesentlichen zwei Aufgaben: Zum einen steuert sie die technische Ausstattung, wie Großrechner, Arbeitsstationen und deren Peripherie, aber auch Aufzüge, Fahrzeuge oder ganze Produktionsanlagen. Zum anderen ermöglicht und definiert sie die Einflussmöglichkeiten des Anwenders auf diese Steuerung.

Programmtechnische Vorgaben üben einen dauerhaften und prägenden Einfluss auf die Arbeitsumgebung und Arbeitsabläufe in einer Vielzahl von Bereichen aus.

Neben der Informationstechnologie ist daher auch in zunehmendem Maße die Rechtswissenschaft weltweit mit Fragestellungen bezüglich Software befasst. Dies beginnt mit der Frage, was Software „eigentlich“ ist, wie sie entsteht, und erstreckt sich über die Konfliktlösung in Projekten aller Art vor allem auf die Entwicklung eines geeigneten rechtlichen Rahmens.

Die vorliegende Untersuchung befasst sich im Schwerpunkt mit den rechtlichen Rahmenbedingungen der Entstehung von Software und geht dabei von der Praxis der Vertragsgestaltung aus.

Ein aus Gründen der Begrifflichkeit hier vorwegzunehmendes Ergebnis der Untersuchung ist das Verwenden der Bezeichnung „Softwareentwicklungsverträge“¹. Der Produktionsprozess von Software ist

¹ Der ursprüngliche Arbeitstitel enthielt noch den Begriff „Softwareerstellungverträge“

aus verschiedenen, unten näher ausgeführten Gründen nicht mit dem anderer Industriegüter zu vergleichen. Ein Gesichtspunkt ist dabei die Ungewissheit der endgültigen Form des Ergebnisses. Ein bis ins Detail beschriebenes Produkt mit einer unumstößlichen Vorgabe hinsichtlich Form, Material, Farbe etc. ist in einer an (vorzugebenden) Trends und (zu weckenden) Bedürfnissen orientierten Produktionsstruktur selten. Im Bereich der Produktion von Software tritt die häufig im Laufe des Projektes sukzessive erfolgende Präzisierung des Ergebnisses hinzu. Es liegt also nicht in der Hand des produzierenden Softwarehauses, Form und Gestalt zu wählen. Das Endprodukt entsteht in einem von Austausch mit dem Softwarebesteller und kontinuierlicher Prüfung geprägten Prozess stufenweise konkretisierenden Vorgehens. Der Begriff der „Entwicklung“ ist daher treffender als der der Erstellung.

Ausgangsbasis bei dieser Entwicklung ist mit fortschreitender informationstechnischer Durchdringung der Gesellschaft zunehmend ein vorhandener Bestand an Hard- und Software. Viele Projekte müssen neue Bestandteile mit alten verknüpfen oder auf Grundlage der alten neue entwickeln. Entwicklung in dem hier verstandenen Sinn umfasst daher auch komplexe Anpassungen².

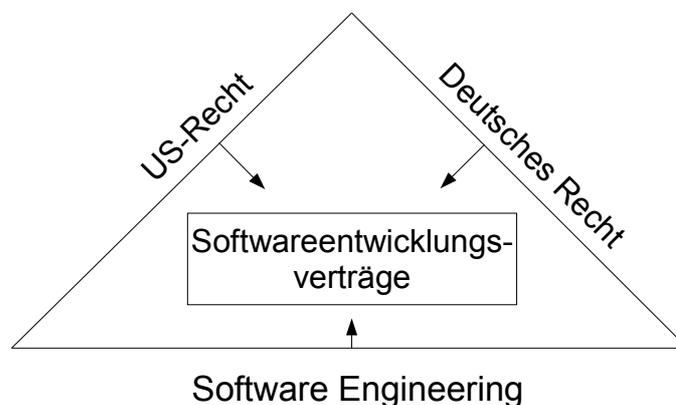
² Keine Unterscheidung findet ferner zwischen der Entwicklung von Standard- und Individualsoftware statt. Deren Dynamik wird, bei unterschiedlicher Akzentuierung im Einzelfall, als vergleichbar vorausgesetzt. Auf notwendige Differenzierungen z.B. bzgl. der Frage, ob die Lieferung des Quellcode Bestandteil der Leistungspflicht ist (vgl. dazu zuletzt Redeker, CR 2005, S. 700 (704)) wird im Rahmen der vorliegenden Untersuchung nicht eingegangen.

1. ALLGEMEINER TEIL

Kapitel 1: Gang der Untersuchung, Bedeutung der IT-Branche und Einführung in die Begriffe Software und Leistungsbestimmung

A. Gang der Untersuchung

Diese Arbeit untersucht Softwareentwicklungsverträge im Hinblick auf die zentrale Frage der Leistungsbestimmung. Sie wird aus drei Perspektiven beleuchtet: Aus der Sicht der Informationstechnologie (Software Engineering – Kapitel 2), aus der Sicht der US-amerikanischen Rechtsordnung (Kapitel 3 und 4) und aus Sicht der deutschen Rechtsordnung (Kapitel 5 und 6). In den Blickpunkt werden dabei für die Praxis vorgesehene Muster und in der Praxis verwendete Verträge gestellt und anschließend miteinander und mit den Anforderungen der Informationstechnologie verglichen (Kapitel 7).



B. Bedeutung der Informationstechnologie

Der Weg in die Informationsgesellschaft kann anhand ausgewählter statistischer Daten nachvollzogen werden. Im Vordergrund stehen dabei die anhand offizieller Statistiken nachvollziehbaren Bestandsdaten an Hardware und der Computernutzung in Unternehmen sowie ausgewählter Kennzahlen

aus dem seit kurzem statistisch separat erfassten Wirtschaftsbereich der Datenverarbeitung und Datenbanken.

Die Ausstattung privater Haushalte stellt sich danach wie folgt dar³:

<i>Gegenstand</i>	<i>1999</i>	<i>2004</i>
PC stationär	43,1	58,7
PC mobil (Notebook, Laptop)	4,9	13,3
Internetanschluss, -zugang	10,7	47,1

In den USA stieg der Anteil der PC-Nutzung zwischen 1998 und 2002 von 458,6 auf 653 pro 1.000 Einwohner, die Ausstattung war im Vergleich zu Deutschland (noch) deutlich besser. Die Zahl der Internet-Hostrechner⁴ wuchs in Deutschland von 1.449.915 in 1998 (USA: 30.489.463) auf 2.603.007 in 2003 (USA: 162.208.993).^{5, 6}

Deutlich mehr als die Hälfte aller Haushalte verfügte in Deutschland in 2004 über einen PC, und davon konnte ein großer Anteil das Internet nutzen. Die erhebliche Wachstumsrate im Bereich der Internetzugänge (und der Hostrechner) zwischen 1999 (1998) und 2004 (2003) kann als Indikator dafür dienen, dass das Internet einen bedeutsamen Wachstumsfaktor in der Branche der Informationstechnologie und vor allem auch der Softwareindustrie darstellt. Ganze Branchenzweige, wie z.B. das Design von Internetpräsenzen (Webdesign), der Einkauf über das Internet (Online-Shopping) oder das Bereithalten von Infrastruktur für Internetpräsenzen (Webhosting) verdanken ihre Existenz der zunehmenden Marktdurchdringung mit dieser, in weitem Umfang auf Software angewiesenen, Telekommunikationstechnologie.

In Unternehmen ergab sich in Deutschland eine Nutzung von Computern in 2003 zu 80 % und in 2004 zu 84 %. Der Anteil der Beschäftigten an Computern stieg von 46 % in 2003 auf 55 % in 2004⁷.

³ Statistisches Jahrbuch 2005 für die Bundesrepublik Deutschland (SJB-D 2005), Wiesbaden, 2005, S. 114. Die Angaben stellen Prozentzahlen (v.H.) dar.

⁴ Laut Statistisches Jahrbuch 2005 für das Ausland (SJB-A 2005), Wiesbaden, 2005: „Computer, die direkt mit dem World Wide Web verbunden sind, die diesbezüglichen Zahlen sollten als Näherungswert betrachtet werden“, S. 301, Fn. 1.

⁵ SJB-A 2005, S. 330.

⁶ SJB-A 2005, S. 300f.

⁷ SJB-D 2005, S. 119.

Auch aus der Anzahl der in der Branche Beschäftigten und der Höhe der erzielten Umsätze kann auf die Bedeutung des Wirtschaftszweiges geschlossen werden. In 2002 wurden im Wirtschaftsbereich der Datenverarbeitung und Datenbanken in Deutschland 42.025 Unternehmen erfasst, die mit 377.908 Beschäftigten einen Umsatz von 56,360 Mrd. € erzielten. Darauf entfallen 26.562 Softwarehäuser mit 228.750 Mitarbeitern und einem Umsatz von 31,229 Mrd. €. In dem Sektor Informations- und Kommunikationstechnologien arbeiteten 2002 insgesamt 1.078.443 Menschen in 57.716 Unternehmen und erzielten einen Umsatz von 270,754 Mrd. €. ^{8,9}

In den USA verdoppelte sich die Anzahl der Beschäftigten im speziellen Bereich des „*Computer programming*“ zwischen 1995 und 2000 von 245.300 auf 518.700. Im „*Online Information Service*“ wuchs die Zahl der Mitarbeiter zwischen 1999 und 2000 von 98.900 auf 177.300¹⁰.

Diese Zahlen belegen die große Bedeutung der Informationstechnologie als Wirtschaftsfaktor sowie die deutlichen Wachstumsraten in der Branche.

C. Erscheinungsformen von Software

Software kann nach vielen Kriterien eingeteilt werden¹¹. Eine gängige Unterscheidung differenziert zwischen Systemsoftware¹² und Anwendungssoftware¹³. Während die Systemsoftware zur Verwaltung und

⁸ SJB-D 2005, S. 118.

⁹ Vergleichszahlen aus dem deutschen Baugewerbe zeigen folgendes Bild: In 2003 erwirtschafteten 14.203 (20.639 in 1999) erfasste Unternehmen mit 744.000 (1.127.000 in 1999) Mitarbeitern einen Umsatz von 85,207 (221,460 in 1999) Mrd. €. SJB-D 2005, S. 367. Sowohl Umsatz als auch Beschäftigung sind also deutlich rückläufig.

¹⁰ Statistical Abstract of the US 2002, 122nd Edition, Issued by the Department of Commerce Dec. 2002, S. 696f. In der „*IT-industry*“ wurden neben *Computer programming* die Kategorien *Prepackaged Software* und *Information Retrieval* erfasst. In diesen letzten beiden Kategorien wuchs die Anzahl der Beschäftigten von 1995 bis 2000 ebenfalls von 180.800 bzw. 56.900 auf 299.900 bzw. 243.000, hatte also ebenfalls einen erheblichen Anstieg zu verzeichnen, der im Bereich des *Information retrieval* besonders auffällig ausfällt.

¹¹ Z.B. nach der Art der Softwareerstellung, dem Grad der Spezialisierung und des Leistungsumfanges oder nach dem Anwendungsbereich, Zilahi-Szabó, M.G., *Wirtschaftsinformatik*, München/Wien 1993, S. 485ff.

¹² Zur Systemsoftware werden Betriebssysteme und Netzwerksysteme gerechnet. Beispiele für Systemsoftware sind Betriebssysteme wie Unix, Linux, Windows, oder Netzwerksoftware wie Novell, vgl. Zilahi-Szabó, a.a.O., S. 485ff.

¹³ Beispiele für Anwendungssoftware sind Textverarbeitung (Word, Word Perfect, Open Office), betriebswirtschaftliche Software (z.B. SAP R3) oder sonstige aufgabenbezogene

Steuerung der internen Vorgänge im Computersystem bzw. im Computernetzwerk dient, löst die Anwendungssoftware Aufgaben, die der Benutzer ihr außerhalb der reinen Interaktion mit dem Rechner oder Netzwerk stellt.

Auf einer anderen Begriffsebene werden Individual- und Standardsoftware unterschieden. Individualsoftware wird speziell für die Belange eines Auftraggebers hergestellt. Eine individuelle Lösung versucht, den spezifischen, situationsbedingten Vorgaben möglichst weitgehend Rechnung zu tragen. Der Anwender definiert durch seine Vorgaben die Entwicklung einer speziell auf seine Bedürfnisse ausgerichteten Software.

Standardsoftware ist demgegenüber die programmtechnische Lösung einer abstrakten Aufgabe (z.B. die Funktion einer Schreibmaschine durch Textverarbeitung zu ersetzen), deren Programmlogik und Verarbeitungsvorgänge fest vorgegeben sind¹⁴. Sie wird nicht für einen einzelnen Auftraggeber, sondern für einen vermuteten oder tatsächlichen Marktbedarf geschaffen. Zentrale Aufgaben wie z.B. im Bereich der Büroanwendungen das Verfassen und Bearbeiten von Texten (Textverarbeitung), das Durchführen von Berechnungen (Tabellenkalkulation), das Speichern, Abrufen, Manipulieren vordefinierter Datensätze (Datenbanken) beruhen auf einem einheitlichen Schema, welches in der Praxis ohne weiteres für viele Anwendungsfälle eingesetzt werden kann. Standardsoftware findet sich überall dort, wo eine Vielzahl von Anwendungsbereichen und damit potentieller Kunden die aufwendige Entwicklung von Software rechtfertigt. Die richtige Einschätzung eines entsprechenden Bedarfs hat zum Entstehen weltweit agierender Konzerne und ganzer Industriezweige geführt.

Ob eine Aufgabenstellung im konkreten Fall durch Standard- oder Individualsoftware gelöst wird, hängt weitgehend von der Wahl und den Ressourcen (v.a. dem Budget) des zukünftigen Anwenders ab.

Programme (von der Vereinsverwaltung über die Maschinensteuerung (egal, ob mobiles Telefon, Aufzug, Gepäckabfertigungs- oder Glasfertigungsanlage) bis hin zur Flug- oder Raumfahrtsteuerung und -überwachung).

¹⁴ Was individuelle Gestaltungsmöglichkeiten jedoch nicht ausschließen muss, wie z.B. die Option der Makroprogrammierung zeigt.

Die in dieser Form gezogene Grenzlinie zwischen Standardsoftware und Individualsoftware wird allerdings aus unterschiedlichen Gründen verwischt. So kann und muss auch Standardsoftware im konkreten Fall an die Bedürfnisse der Anwender angepasst werden. Dies geschieht über die Einstellung von Optionen (sog. Customizing), wie in einem einfachen Fall der Auswahl der Sprache oder Darstellung. Bei umfangreicheren Anwendungen kann die Anzahl der einzurichtenden Parameter durchaus im vierstelligen Bereich liegen. Die Verknüpfung von Anwendungen erfordert darüber hinaus eine Bearbeitung, u.U. eine Neuerstellung von Schnittstellen, um z.B. eine Übernahme von Daten zu ermöglichen (etwa die Nutzung der Adresdaten aus einer Datenbank für ein Rundschreiben). Unter anderem zu diesem Zweck beinhaltet Standardsoftware nicht selten eine eigene Programmiersprache, die individuelle Anpassungen ermöglicht.

Individualsoftware bedient sich auf der anderen Seite in zunehmendem Maße bereits vorhandener Programmelemente (Objekte). So machen sich Programmiersprachen, die konzeptionell in der Verknüpfung immer wieder auftretender Programmelemente bestehen, ein Baukastenprinzip zunutze. Die sog. objektorientierten Programmiersprachen z.B. versuchen, auf diesem Weg der Kombination bereits entwickelter Lösungen zum Ziel zu gelangen.

Es tritt die Möglichkeit einer Verbindung von Standardsoftware mit Individualsoftware hinzu, wie sich am derzeit häufig anzutreffenden Beispiel des Verknüpfens einer individuell programmierten Website mit Hintergrundanwendungen wie etwa einer Datenbank zeigen lässt.

Hinsichtlich ihrer Größe und der Gliederung in Teile können Softwareprojekte wie folgt eingeteilt werden¹⁵:

Organisationsgröße	Klein	Mittel	Groß	Sehr groß
Entwickler	< 10	> 10	> 300	> 1000
Anzahl (Teil-) Projekte	< 4	> 4	> 50	> 130
Entwickler/Projekt	4	5	6	8

¹⁵So bei Spitta, Thorsten, Scriptum Informationsmanagement, S. 5, abrufbar (zuletzt 12.12.05) unter <http://www.wiwi.uni-bielefeld.de/~spitta/download/IM3.pdf>; die Zahlenangaben sollen und können lediglich einen Eindruck von der unterschiedlichen Aufgabenstellung abhängig von der Anzahl der beteiligten Personen geben.

Beispiele für Großprojekte sind die Entwicklung eines Navigationssystems für Kfz oder die Entwicklung eines Warenwirtschaftssystems für eine Handelskette. Mittlere Projekte sind die Entwicklung einer betriebswirtschaftlichen Software für ein mittelständisches Unternehmen mit mehreren Filialen. Ein kleines Projekt stellt beispielsweise die Entwicklung der Website für den Fachbereich einer Universität dar¹⁶.

D. Die Bedeutung der Leistungsbestimmung

Im Rahmen der Softwareentwicklung wird üblicherweise von einem Projekt gesprochen. An diesem Projekt können eine Vielzahl von Personen partizipieren¹⁷. Strukturell und häufig vertraglich sind zwei Seiten beteiligt: Das Softwarehaus und der Softwarebesteller. Auf der einen Seite wird durch das Softwarehaus Wissen der Informationstechnologie repräsentiert. Auf der anderen Seite hat sich der vermutete oder tatsächliche Bedarf für die Lösung oder Unterstützung der Lösung einer Aufgabenstellung mit Mitteln der Informationstechnologie gezeigt. Daraus entwickelt sich ein „Projekt“, für das die Beteiligten früher oder später nach juristischer Begleitung suchen. Was sind nun die Interessen der Beteiligten? Eine allgemein zutreffende Antwort darauf gibt es nicht; zu unterschiedlich sind die Ausgangssituationen und die Wünsche und Motive der Menschen. Abstrakt ließe sich immerhin formulieren: Die Beteiligten erwarten von der Tätigkeit der Juristen eine Förderung des Projektes, möglicherweise wünschen sie, infolge einschlägiger Erfahrungen, auch lediglich den schädlichen Einfluss der rechtlichen Profession möglichst gering und unter Kontrolle zu halten¹⁸. Wird eine dauerhafte, in aller Regel auch heute noch schriftliche, vertragliche Fixierung für nötig befunden, so bezieht sie sich auf zwei Regelungskreise. Der erste betrifft, gewissermaßen rückwärts gerichtet, das In-Worte-Fassen des aktuellen Projektstands, also die Darstellung des status quo. Der zweite

¹⁶ Die Beispiele können natürlich nicht repräsentativ sein. Je nach Anforderungen kann auch die Aufgabenlösung für das mittelständische Unternehmen zu einem Großprojekt, die Website zu einem mittelgroßen Projekt werden.

¹⁷ Wie z.B. im in die Schlagzeilen geratenen Toll-Collect-Projekt, einem großen Industriekonsortium mit aktueller Beteiligung der DaimlerChrysler Financial Services AG (45%), der Deutschen Telekom AG (45%) und der französischen Cofiroute S.A. (10%), http://www.toll-collect.de/faq/tcrdifr004-1_unternehmen.jsp, zuletzt abgerufen am 05.12.05.

¹⁸ Als Juristen sollten wir ehrlich genug sein, dieser bisweilen durchaus anzutreffenden Haltung mit offenen und selbstkritischen Augen entgegenzusehen.

betrifft, vorwärtsgerichtet, die antizipierte Zukunft des Projekts. Die Aufgabe der rechtlichen Berater ist es mithin, eine Form zu gestalten, in der die Beteiligten das bisher Erreichte und das angestrebte Ziel wiederfinden. Die Leistungsbestimmung, wie sie hier verstanden wird, betrifft beide Regelungskreise. Ihre Aufgabe ist es, (1) das Ziel festzulegen, (2) den Weg dorthin aufzuzeigen, was (3) eine Standortbestimmung voraussetzt. Dies soll auf Basis der Softwareentwicklung, wie sie sich aus der Sicht der Informationstechnologie darstellt, geschehen.

Zunächst ist also - aus softwaretechnischer Sicht - die Frage zu beantworten, unter welchen Grundvoraussetzungen ein Softwareprojekt durchgeführt wird und welche Fragestellungen und Rahmenbedingungen sich daraus für die rechtliche Perspektive ergeben.

Kapitel 2: Softwareentwicklung in der Praxis

A. Theorie des Softwareengineering

I. Einführung

Die Bestimmung der Leistungspflicht, die für den Vertragsjuristen einen zentralen Ausgangspunkt für den Vertragsentwurf darstellt, ist aus Sicht der Wissenschaft, die sich mit der Softwareentwicklung befasst, und aus Sicht der Systementwickler ein hochkomplexer Prozess¹⁹.

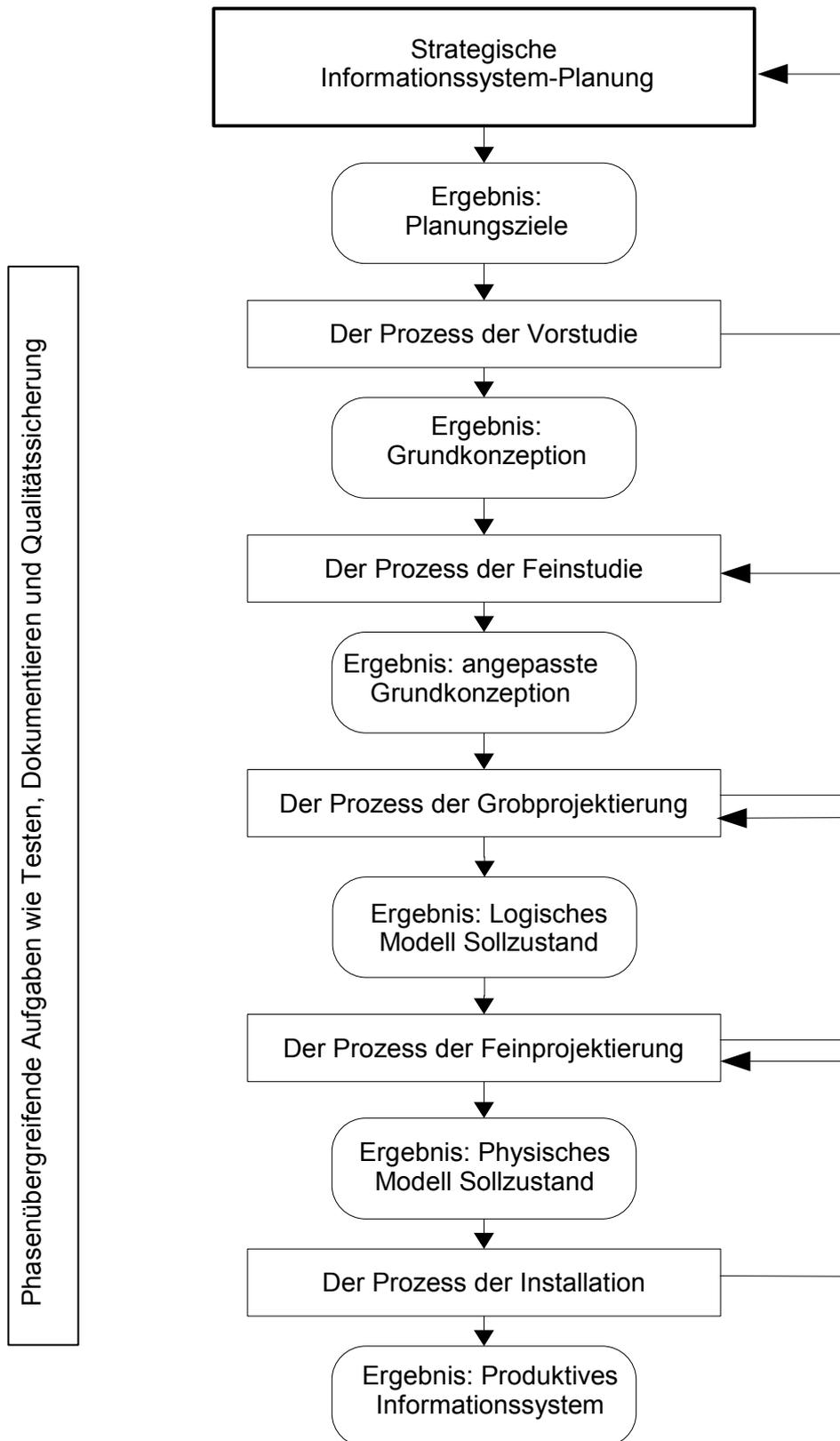
Softwareentwicklung vollzieht sich danach in bestimmten, definierten Schritten²⁰. Die Planungsphase, in der die Leistungsbestimmung ihren Ausgangspunkt hat, vollzieht sich ihrerseits in verschiedenen Abschnitten, die das Ziel der Softwareentwicklung und den Weg dorthin in eine zunehmend genauere Form fügen.

Heinrich²¹ unterteilt die Systemplanung wie aus der folgenden Grafik ersichtlich.

¹⁹ Etzel/Heilmann/Richter definieren IT-Projekte als „*einmalige Vorhaben mit hoher Komplexität, die innovatives Vorgehen erfordern, deren Start und Ende terminiert, deren Ressourcen begrenzt und die mit Risiken verbunden sind*“, IT-Projektmanagement, Heidelberg, 2000, S. 3; die Unbeherrschbarkeit der Komplexität der Softwareentwicklung wird besonders hervorgehoben von Eschbach, Software nach Maß, Haar, 1992, S.140; Love, Tom, Tücke des Projekts, Hannover, 1994, S. 33; Zehnder, Informatik-Projektabwicklung, 2. Aufl., Stuttgart, 1991, S. 119.

²⁰ In diesem Zusammenhang wird häufig auch von einem Software-Lebenszyklus, Dumke, Software-Engineering, 3. Aufl., Braunschweig Wiesbaden, 2001, S. 17ff, oder auch Software-Lebenslauf, Frühauf/Ludewig/Sandmayr, Software-Prüfung, Stuttgart, 1991, S.14ff, gesprochen.

²¹ Heinrich, Lutz J., Systemplanung, 6. Aufl., München/Wien, 1994, S. 43.



Bemerkenswert ist seine Feststellung zu diesem Schema, wonach das Produkt der Systemplanung, also das Ergebnis der Entwicklung, zu Beginn

des Planungsprozesses weitgehend unbekannt ist²². Die Beteiligten haben bei dieser Sichtweise also allenfalls ein abstraktes Ziel (eine bestimmte Funktionalität) und eine grobe Route (die theoretische Abfolge der Softwareentwicklung) vor Augen. Diese Problematik von Planbarkeit in der Softwareentwicklung wird auch von zahlreichen anderen Autoren betont²³. Anders als etwa bei der Erstellung eines Bauwerkes gibt es kein physisches, also sinnlich erfahrbares Modell, an dem sich Auftraggeber und Softwarehaus orientieren können²⁴. Etwas anders und provokativer ausgedrückt: Die Beteiligten begeben sich auf eine Reise, bei der sie weder Ziel noch Route kennen.

In dem Bemühen, diese Aufgabe trotzdem zu bewältigen, haben sich zur Beschreibung des Ablaufes der Softwareentwicklung in der Literatur zahlreiche Modelle²⁵ herausgebildet.

Sinn dieser Softwareentwicklungsmodelle ist die Reduzierung von Komplexität. Eine unüberschaubare Aufgabenstellung wird in einzelne Abschnitte und/oder Teilprojekte gegliedert, in denen u.a. die Grundlagen für die Entscheidungen grundsätzlicher Bedeutung, wie beispielsweise die Erstellung eines Pflichtenheftes, geschaffen werden²⁶.

II. Modelle für die Erstellung von Software

Eine Vielzahl von Modellen hat den Versuch unternommen, die Phasen der Entwicklung in eine logische Abfolge zu bringen. Im Ergebnis besteht weitgehend Übereinstimmung hinsichtlich der erforderlichen Arbeitsschritte, wie sich anhand zweier Phasenmodelle²⁷ zeigen lässt.

²² Heinrich, a.a.O., S. 43.

²³ Zehnder, a.a.O., S. 59f; Eschbach, a.a.O., S.140. Love schreibt dazu instruktiv: „*Trotz genauester Planung geschieht immer wieder das Unerwartete.*“ a.a.O., S. 189.

²⁴ Was eine Folge der „*Unsichtbarkeit und Fügbarkeit*“ – so Brooks, Vom Mythos des Mann-Monats, Bonn, 1987, S. 102 - von Software ist.

²⁵ Eine anschauliche Übersicht über 12 sequentielle Vorgehensmodelle findet sich bei Kargl, Herbert, Management und Controlling von IT-Projekten, München Wien, 2000, S. 75.

²⁶ Hansen, Hans Robert/Neumann, Gustav, Wirtschaftsinformatik I, Grundlagen betrieblicher Informationsverarbeitung, 8. Aufl., Stuttgart, 2001, S. 204f.

²⁷ Ausschnitt aus einer umfangreichen Übersicht nach Kargl, a.a.O., S. 75.

Planung
Definition
Entwurf
Implementierung
Abnahme/Einführung
Wartung und Pflege

Balzert 1982

Projektdefinition
Ist-Analyse
Sollkonzept
Realisierungsplanung
Software-Entwicklung
Systemeinführung

Walter 1992

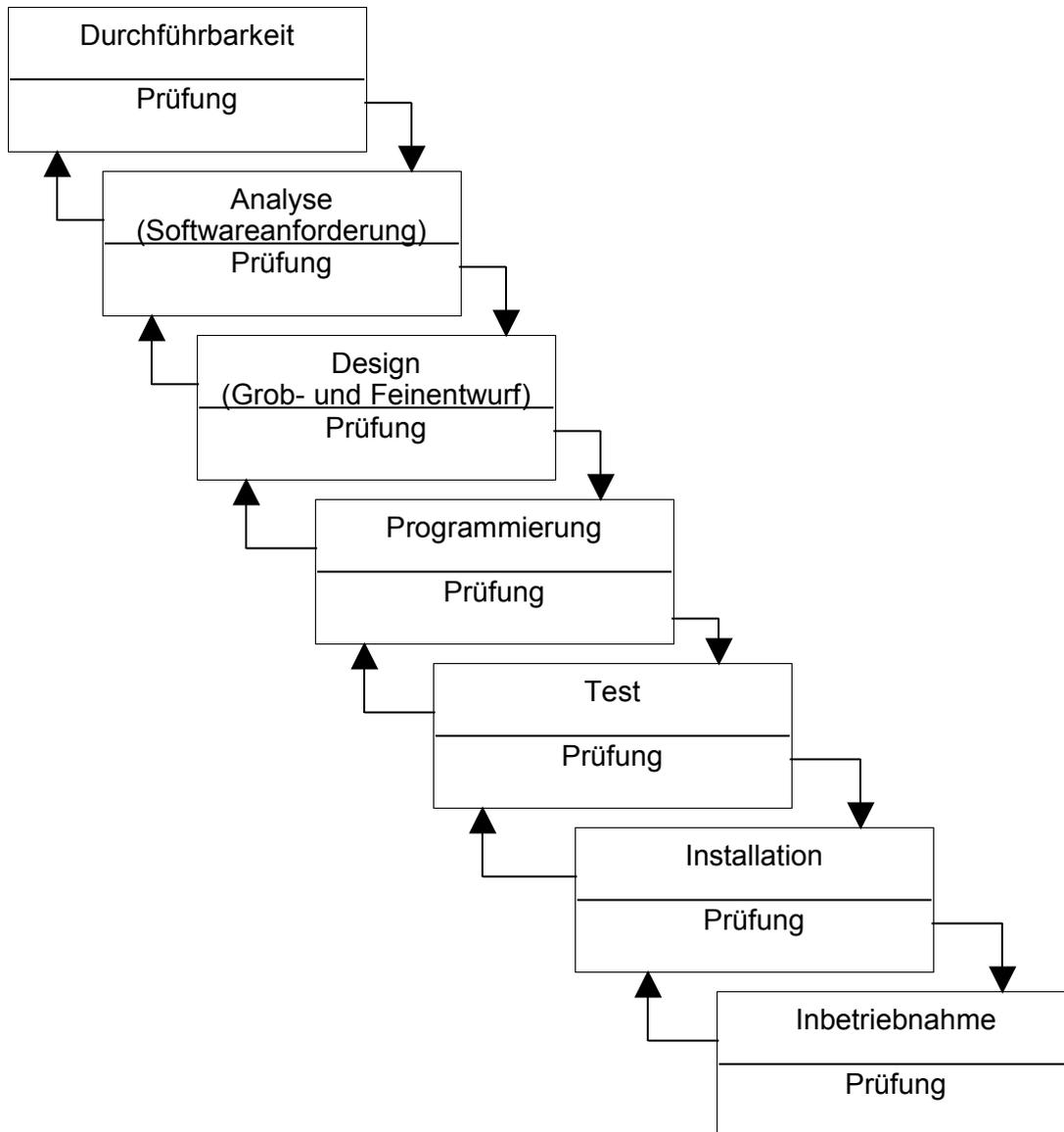
In der Praxis zeigte sich, dass die einzelnen Abschnitte nicht strikt voneinander zu trennen sind. Sie überlappen sich vielmehr und müssen wegen auftretender Fehler und Änderungen in der vorgelagerten Phase erneut durchlaufen werden²⁸.

Diesem Sachverhalt tragen weiterentwickelte Modelle konzeptionell Rechnung. Im Folgenden sollen drei dieser Modelle näher betrachtet werden.

Im sog. Wasserfallmodell²⁹ ist als Ende jeder Phase eine Prüfung und im Fall des Auftretens von Fehlern ein wiederholtes Durchlaufen der Phase vorgesehen. Im Unterschied zu den vorangegangenen Modellen findet hier also im Anschluss an jede Phase eine Validierung statt, und die mögliche Rückkehr zu einem vorgelagerten Entwicklungsstadium ist bereits Bestandteil der Betrachtungsweise.

²⁸ Hansen/Neumann, a.a.O., S. 207.

²⁹ Das Wasserfallmodell repräsentiert eine auch heute noch verwandte Methode, die bereits Anfang der 70er Jahre eingeführt wurde, McManus / Wood-Harper, Information Systems, S. 77, die einzelnen Schritte werden mit unterschiedlichen Schwerpunkten und Bezeichnungen dargestellt, vgl. Mall, Software Engineering, S. 19; die grundsätzliche Struktur ist in der folgenden Grafik dargestellt.

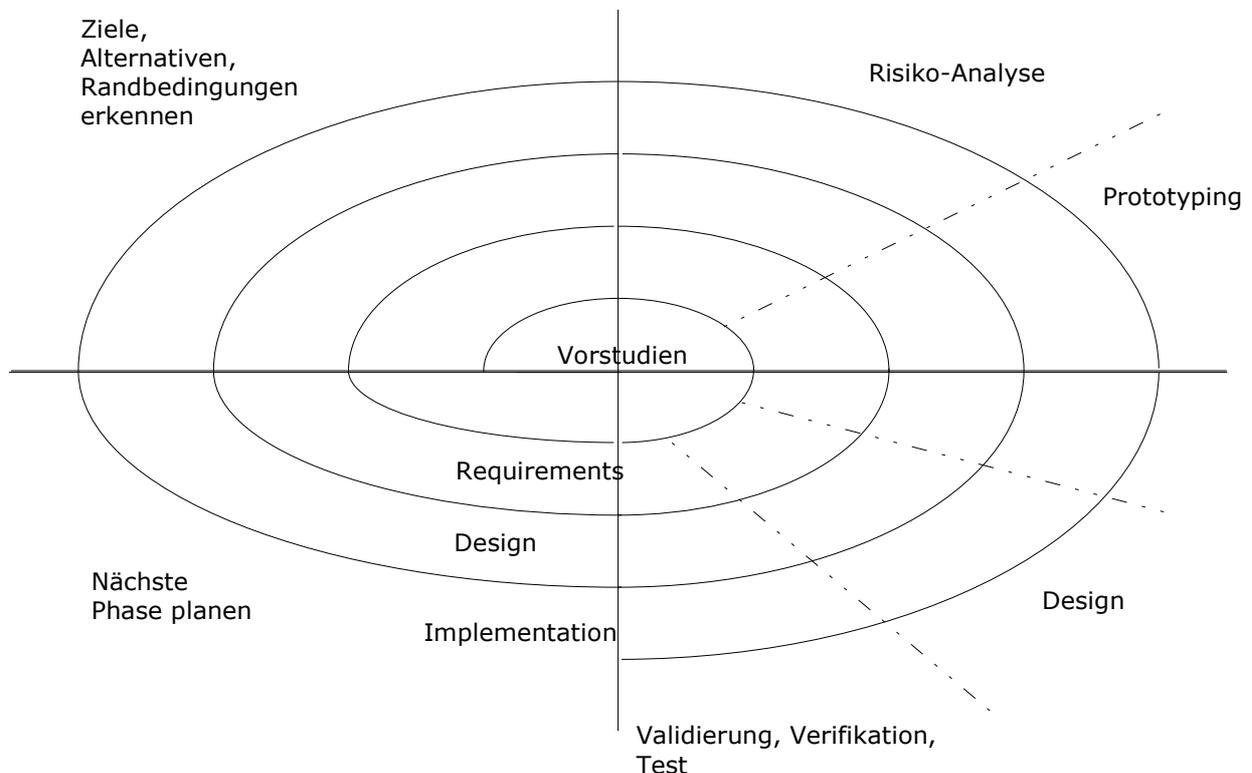


Das Wasserfallmodell geht von einem konsequenten Abarbeiten der einzelnen Abschnitte aus, wobei jede Stufe so lange durchlaufen wird, bis ein vollständiges und fehlerfreies Ergebnis vorliegt. Der Rücksprung erfolgt dabei (nur) auf die jeweils vorgelagerte Stufe.

Im Spiralmodell³⁰ werden in jeder Phase von der ersten Formulierung der Anforderungen bis zum produktiven Einsatz definierte Zyklen durchlaufen:

³⁰ Dieses Modell wurde 1988 von Barry Boehn mit dem Ziel vorgestellt, die Risiken der Softwareentwicklung zu reduzieren, McManus / Wood-Harper, a.a.O., S. 83.

1. Festlegung von Zielen, Alternativen und Randbedingungen
2. Risikoanalyse und ggf. Entwurf von Prototypen
3. Entwurf (Design) und Prüfung
4. Implementation und Planung der nächsten Phase



Das Spiralmodell ist eine Weiterentwicklung des Wasserfallmodells, wobei durch die Risikoanalyse und das Prototyping bei jedem Schritt die Entwicklung von Kosten und Zeitplan besser im Auge behalten werden kann. Die unterschiedliche Terminologie sollte freilich nicht darüber hinwegtäuschen, dass die zeitliche Abfolge der einzelnen Schritte von der Projektdefinition (hier Vorstudie) bis zur Systemeinführung (hier Installation) unverändert geblieben ist. Das Spiralmodell gibt jedoch bereits durch seine graphische Gestaltung einen Hinweis auf die zyklische Entwicklung innerhalb des Projektes. Es ist durch eine Vorgehensweise geprägt, die als *Iteration* bezeichnet wird (ein Verfahren zur Verbesserung von Näherungslösungen durch ständiges Wiederholen des gleichen Ablaufes). Nicht jeder weitere Schritt führt dabei zwangsläufig näher zum Ziel (Fertigstellung der Software). Es können vielmehr unvorhergesehene Risiken

(wie Kostensteigerungen) auftauchen oder Umstellungen in der weiteren Planung (wie eine Verlängerung der Projektdauer) erforderlich werden. Es kann sich also herausstellen, dass der aktuell zu veranschlagende Aufwand und angestrebte Fertigstellungszeitpunkt deutlich von der ursprünglichen Prognose abweichen und das Ziel damit in weite Ferne rückt.

Besondere Aufmerksamkeit verdient das im Spiralmodell explicit vorgesehene sog. Prototyping³¹. Bei dieser Vorgehensweise wird bereits zu einem frühen Zeitpunkt ein Prototyp entwickelt und in Abstimmung mit den Anwendern stufenweise verfeinert. Dies führt naturgemäß dazu, dass die Phasen Entwurf und Implementation eng miteinander verzahnt werden und teilweise parallel ablaufen³².

Das letzte hier darzustellende Modell ist das V-Modell.

Es ist vom Ende der 80er bis zum Anfang der 90er Jahre, initiiert vom Deutschen Bundesministerium für Verteidigung, zur Optimierung der Softwareentwicklung als Ergebnis eines Forschungs- und Technologieprojektes entstanden³³. Die Entwicklung wurde in Zusammenarbeit mit dem Bundesministerium des Inneren (BMI) und den Verbänden der deutschen Industrie mit dem Ziel betrieben, die Softwarekosten einzudämmen, die Qualität der Software zu verbessern und die Abhängigkeit des Softwarebestellers vom Softwarehersteller zu vermindern³⁴. Im Jahr 1992 übernahm das BMI das Verfahren für den zivilen Verwaltungsbereich und machte es damit zu einem Standard für den öffentlichen Bereich. Es hat auch darüber hinaus in der Privatwirtschaft Anerkennung gefunden, wo es als Entwicklungsstandard in vielen Unternehmen Verwendung findet³⁵.

³¹ Dieser Grundgedanke findet sich bereits bei Brooks, Mythos, S. 102, Prototyping wird insbesondere bei Projekten, die der Benutzerschnittstelle und/oder der Interaktion mit dem Anwender große Bedeutung beimessen, als sinnvoll angesehen, vgl. McManus/Wood-Harper S. 84.

³² Suhl, in: Fischer/Herold/Dangelmaier/Nastansky/Suhl, Bausteine der Wirtschaftsinformatik, 3. Aufl., Berlin, 2002, S. 375.

³³ Bröhl/Dröschel, Das V-Modell, 2. Aufl., München, 1995, S. 150.

³⁴ Bröhl/Dröschel, a.a.O., S. 16.

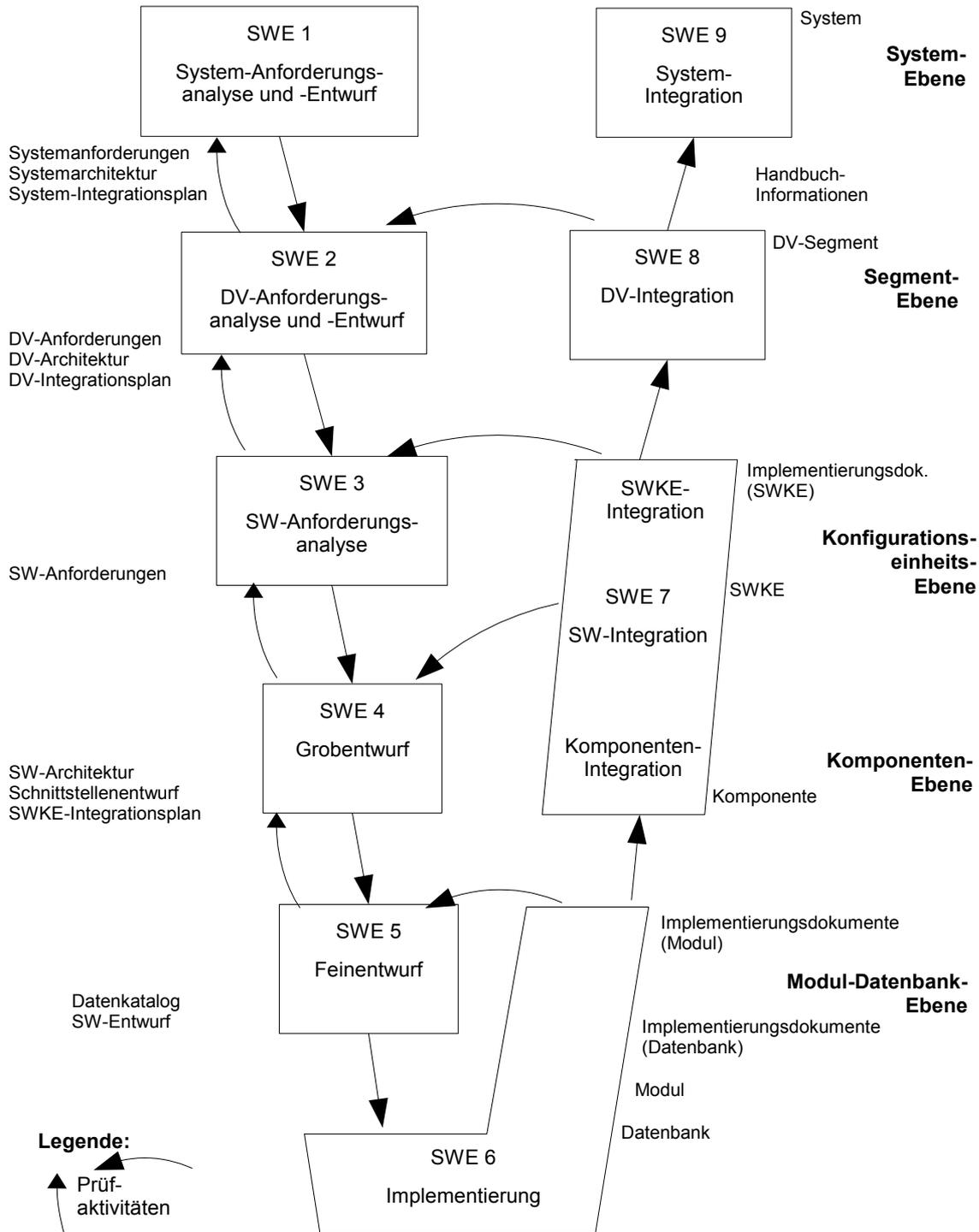
³⁵ Hansen/Neumann, a.a.O., S. 211, eine Liste offizieller Verwender des V-Modells aus Deutschland, Österreich und der Schweiz findet sich unter www.v-modell.iabg.de/VM-Anwender.html.

Das V-Modell ist ein sehr umfangreich und ausführlich dokumentiertes Modell³⁶. Die Softwareentwicklung stellt darin einen Teil in dem Gesamtsystem des Entwicklungsstandards für IT-Systeme dar³⁷. Auch hier wird die Komplexität durch Zerlegung in einzelne Anschnitte und verschiedene Aufgabenbereiche reduziert³⁸. Der Aufgabenbereich Softwareentwicklung (auch als Submodell SWE bezeichnet) ist in neun Schritte gegliedert.

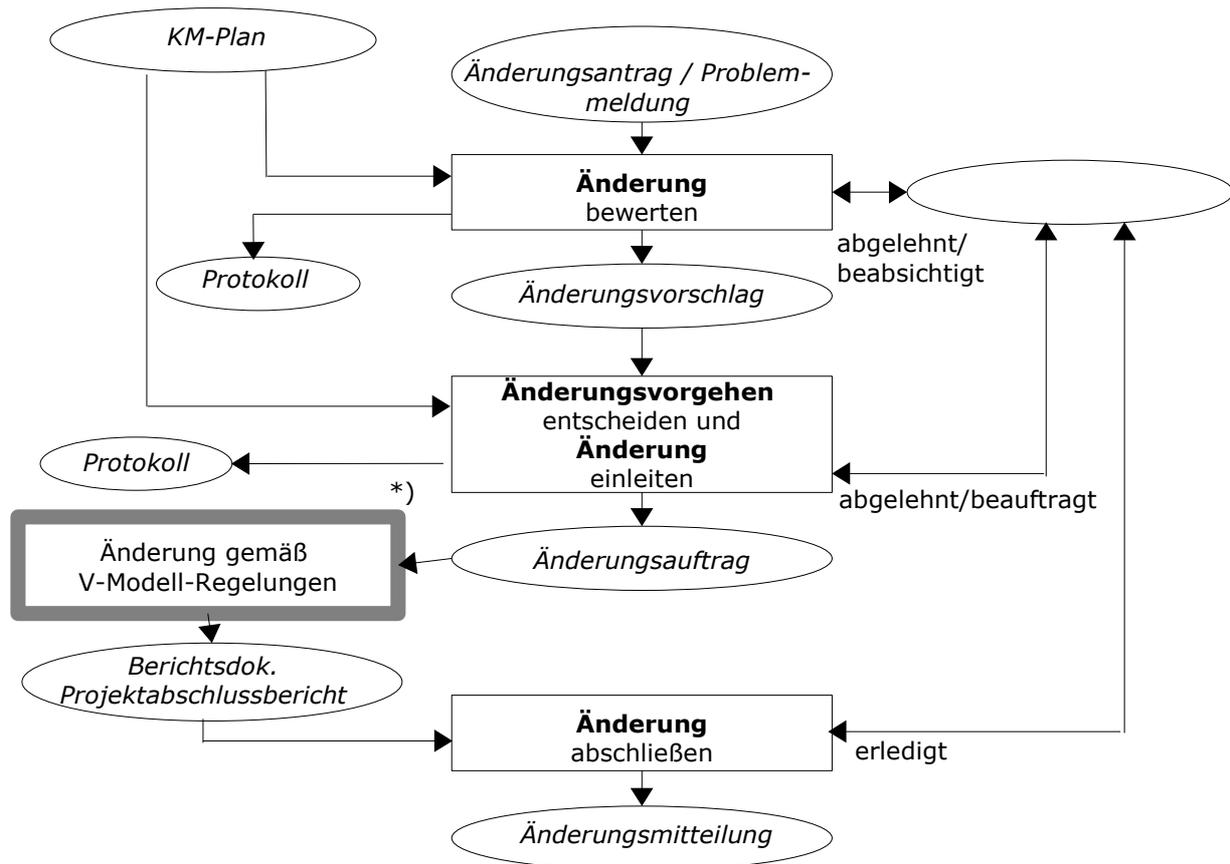
³⁶ Die offizielle Beschreibung erstreckt sich über mehrere hundert Seiten, siehe den Anhang in Bröhl/Dröschel, a.a.O., Fn. 9.

³⁷ Im V-Modell 97, der 1997 fertiggestellten und mittlerweile in aktualisierter Form vorliegenden offiziellen Anleitung zur Durchführung von IT-Projekten, werden Projektmanagement, Qualitätssicherung, Konfigurationsmanagement und Softwareentwicklung als Tätigkeitsbereiche unterschieden.

³⁸ Vier Aufgabenbereiche werden unterschieden: Das Produktmanagement (PM), das Konfigurationsmanagement (KM), die Qualitätssicherung (QS) und die eigentliche Softwareentwicklung (SWE). Das PM plant und steuert den Entwicklungsprozess, das KM verwaltet die fertigen (Zwischen-) Produkte, Konfigurationen und Rechte, übernimmt das Change-Management, die Datensicherung und ein Berichtswesen (Bröhl/Dröschel, a.a.O., S. 27/28), die QS legt den organisatorischen und abwicklungstechnischen Rahmen fest, prüft, ob vorgegebene Methoden eingehalten werden, bereitet die Produktprüfung vor und führt sie durch, trifft die Durchführungsentscheidung anhand der Frage, ob Kosten und Termine eingehalten wurden und Vorprodukte vorliegen, sowie dokumentiert die Prüfungen und wertet sie aus (Bröhl/Dröschel) a.a.O., S. 25/26).



Die Grafik macht auch hier deutlich, dass einerseits eine klare zeitliche Reihenfolge gegeben ist, andererseits die Ergebnisse immer wieder an den Vorgaben und den vorangegangenen Stufen zu messen sind (ständiger Prüfprozess). Änderungen sind als zentraler und selbstverständlicher Bestandteil des Aufgabenbereiches Konfigurationsmanagements (Submodell KM) im Abschnitt Änderungsmanagement erfasst.



*) vollständiges Änderungsprojekt oder definierte Aktivitätenfolge

Änderungen können dazu führen, dass der Produktzyklus teilweise oder vollständig hinsichtlich einzelner Teile oder in seiner Gesamtheit neu durchlaufen werden muss. Das V-Modell versucht in möglichst vollständiger Weise, alle Aspekte der Entwicklung theoretisch zu erfassen und in ein formalisiertes Verfahren zu integrieren.

Theorie und Praxis des Softwareengineering bringen naturgemäß immer neue Modelle und Vorgehensweisen hervor³⁹. Soweit ersichtlich sind derzeit aber keine durchgreifend neuen Ansätze für die vorliegende Untersuchung von Bedeutung⁴⁰.

³⁹ Für die Softwareentwicklung sind im Laufe der letzten 50 Jahre mehr als 300 Normen von 50 verschiedenen Organisationen geschaffen worden, siehe bei Thaller, Software- und Systementwicklung, Hannover 2002, S. 29.

⁴⁰ Modelle wie Rapid Application Development (RAD) oder Dynamic Systems Development Method (DSDM), vgl. McManus / Wood-Harper S. 87ff, können hier außer Betracht bleiben, da sie gegenüber den betrachteten Modellen keine für den Untersuchungsgegenstand maßgeblichen Neuerungen enthalten.

III. Für den Untersuchungsgegenstand besonders bedeutsame Aspekte der Softwareentwicklung

1. Der lange Weg zum Pflichtenheft – Requirements Engineering

a) Einführung

Der Begriff des Pflichtenhefts hat sich als Bezeichnung für die detaillierte Beschreibung der Anforderungen an ein Produkt (Hardware und Software) im Bereich der Praxis der Softwareentwicklung durchgesetzt⁴¹. Auch in der (juristischen) Fachliteratur wird der Begriff allgemein als Definition von Vorgaben und Anforderungsspezifika für Software verstanden⁴².

Die Einführung neuer oder Änderung bestehender Software setzt im Rahmen der Planung modellunabhängig zunächst eine Erfassung der Einsatzbedingungen (eine so genannte Ist-Analyse⁴³) voraus. Das Softwarehaus muss also den Ausgangspunkt für die Entwicklungsarbeit, den Punkt, von dem aus die Reise beginnen soll, kennen. Je komplexer die vorgefundene Situation, um so umfangreicher fällt demgemäß eine Ist-Analyse aus. Heute sind kaum mehr Situationen denkbar, in denen ein Softwarehaus auf „EDV-freie“ Rahmenbedingungen trifft⁴⁴. Die vorhandenen Ressourcen müssen daher auf weitere Verwertbarkeit und, falls dies zutrifft, auf die Notwendigkeit von Schnittstellen mit der zu erstellenden Software geprüft werden.

⁴¹ Vgl. Brockhaus, Computer und Informationstechnologie, Mannheim Leipzig, 2003, unter *Pflichtenheft*; Balzert, Lehrbuch Grundlagen der Informatik, 2. Aufl., München, 2005, definiert das Pflichtenheft als „*Festlegung der Anforderungen an ein Software-System in verbaler, d.h. umgangssprachlicher Form*“; Feretti, Wörterbuch der Datentechnik, Berlin Heidelberg New York, 1996, setzt das Pflichtenheft mit „Spezifikationen“ gleich und definiert sie schlicht als „*vorgegebene technische Bedingungen*“. Grieser/Irlbeck, Computer Lexikon, 2. Aufl., München, 1995, definieren das Pflichtenheft als „*Schriftstück, das vom Softwarehaus in Zusammenarbeit mit dem Kunden verfasst wird und die detaillierte Beschreibung der gewünschten Fähigkeiten enthält.*“

⁴² Vgl. Schaub, CR 1993, S. 329, die darauf hinweist, dass der ebenfalls verwandte Begriff der *Leistungsbeschreibung* keine sachlichen Unterschiede aufweist.

⁴³ Grupp, Bruno, EDV-Pflichtenheft zur Hardware- und Softwareauswahl, 2. Aufl., Köln, 1991, S. 29ff, vgl. auch Dumke, der von einer „Anforderungsanalyse“ spricht, sie aber als eine der Pflichtenhefterstellung nachgelagerte Phase ansieht, a.a.O., S. 32ff, die Terminologie ist vielfältig (Situationsstudie, Vorstudie u.ä.), vgl. Kargl, a.a.O., S. 75.

⁴⁴ Etzel/Heilmann/Richter sprechen von einem „zwangsläufigen Nebeneinander von Informations- und Kommunikationssystemen unterschiedlichen Alters und Stils“, a.a.O., S. 14.

Die mögliche Route kann also wahrscheinlich nicht völlig frei gewählt werden. Bestehende Handicaps oder Vorgaben schränken die Wahl des Weges ein⁴⁵.

Im nächsten und für diese Untersuchung besonders interessanten Schritt ist die Vorstellung des Softwarebestellers vom Projektziel zu ermitteln, zu überprüfen und in ein Anforderungsprofil zu kleiden⁴⁶. Das sog.

Requirements Engineering beschreibt den „systematischen Weg von der Projektidee über die Ziele zu einem vollständigen Satz von Anforderungen“⁴⁷.

Am Ende dieser Phase kann somit eine Definition der zu erbringenden Leistung stehen. Diese dient als Fahrplan oder Pflichtenheft, enthält idealtypisch die „Gesamtheit der für eine bestimmte Problemlösung maßgebenden Zielvorstellungen, Randbedingungen und Bewertungskriterien“⁴⁸ und stellt das Grundgesetz⁴⁹ für das weitere Vorgehen dar⁵⁰. Die aus der Wortwahl ersichtliche Aufmerksamkeit in dieser Entwicklungsphase scheint aus Sicht der Praxis nicht deutlich genug hervorgehoben werden zu können⁵¹. Franz⁵² stellt zur Beschreibung des Inhaltes eines Pflichtenheftes pointiert fest: „Zum Schluss sollte unter allen beteiligten Entscheidungsträgern völlige Einigkeit darüber bestehen, dass das Vorgelegte wirklich das ist, was erreicht werden soll“⁵³.

⁴⁵ Dies kann soweit gehen, dass der Aufwand für die Bereitstellung von Altdaten den übrigen Projektaufwand übersteigt, siehe Zehnder, a.a.O., S. 107.

⁴⁶ Auch für die Bezeichnung dieses Abschnittes der Entwicklung gibt es keine einheitliche Terminologie, es ist von Systemspezifikation (vgl. auch Maymon/Cave, Leitfaden des Software-Projektmanagements, Wiesbaden, 1988, mit anschaulicher Grafik zur Verdeutlichung der Zusammenhänge auf S. 46), Sollkonzept (Suhl/Blumstengel, in: Fischer/Herold/Dangelmaier/Nastansky/Suhl, a.a.O., S. 333 und S. 363ff) oder auch Anforderungsspezifizierung (vgl. die Darstellung in Form eines Leitfadens bei Kargl, a.a.O., S. 198ff) die Rede, teilweise ist eine Anforderungsdefinition im Modell gar nicht explizit vorgesehen (vgl. etwa bei Schwickert, Axel, Web Site Engineering, Stuttgart Leipzig Wiesbaden, 2001, S. 159).

⁴⁷ Rupp, Requirements-Engineering und -Management, 2. Aufl., München Wien, 2002, S. 13.

⁴⁸ nach der Definition von Zehnder a.a.O., S. 40.

⁴⁹ Franz, Techniken der modernen Software-Entwicklung, München, 1991, S. 141.

⁵⁰ Kritisch insoweit Eschbach, der das Pflichtenheft als Projektplan für ungeeignet hält, a.a.O., S. 151.

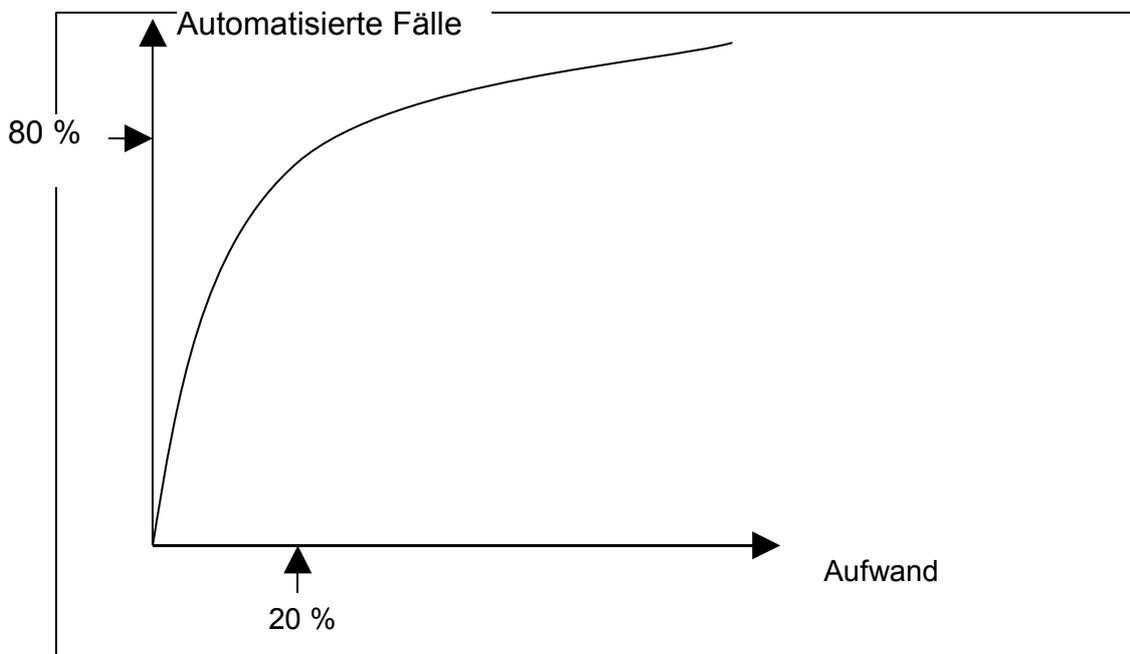
⁵¹ So formuliert Love: „*Beginnen Sie niederzuschreiben, was Sie als Anforderungen verstanden haben. Lassen Sie diese Anforderungen vom Auftraggeber bestätigen, bevor Sie auch nur einen weiteren Arbeitsschritt unternehmen. Es ist unglaublich schwierig, selbst einfache Konzepte einem anderen Menschen genau mitzuteilen. Es soll niemand die seltsamen zwischenmenschlichen Missverständnisse unterschätzen.*“ a.a.O., S. 120, vgl. auch Maymon/Cave, a.a.O., S. 85.

⁵² a.a.O., S. 66.

⁵³ Einige statistische Angaben zu den Erfahrungen in der Praxis, siehe unten: Kritische Faktoren in der Softwareentwicklung.

b) Aufwand

Unabhängig von der Größe des Projektes erfordert die Planungsphase mit dem (Teil-) Ergebnis des Entwurfes eines Pflichtenheftes erheblichen Personaleinsatz. Der Aufwand für die Planung liegt deutlich über dem der Implementation, also der Programmierstätigkeit⁵⁴. Bemerkenswert ist dabei die unter dem Stichwort Pareto-Prinzip⁵⁵ (auch) in der Softwareentwicklung bekannte Verteilung des Aufwands. Nach diesem, auch als 80/20 Regel bezeichneten Prinzip, lassen sich bei der Automatisierung vieler ähnlicher Fälle mit 20% des Automatisierungsaufwandes 80% aller Fälle (nämlich die einfacheren) behandeln, während für die komplizierteren 20% der Fälle 80% des Automatisierungsaufwandes anfallen⁵⁶.



Wenn der Softwarebesteller also einen Automatisierungsgrad von mehr als 80 % anstrebt, d.h. mit der vorgestellten Softwarelösung mehr als 80 % der

⁵⁴ Vor Beginn der Programmentwicklung sind nach Maymon/Cave bereits 50% des Gesamtaufwands verbraucht; a.a.O., S. 73 und 77; Love gibt den Aufwand für die Codierung mit „höchstens 20%“ an, a.a.O., S. 199; nach Eschbach entfallen auf Analyse und Planung, Kodierung und Test sowie Implementierung jeweils 10 % während 70 % für die Wartung, Pflege und Erweiterung aufgewendet werden müssen, a.a.O., S. 81.

⁵⁵ Vilfredo Pareto lebte im 19. Jahrhundert und war Professor für politische Ökonomie an der Universität von Lausanne. Er erkannte, dass in vielen Märkten überall auf der Welt ein Großteil der Aktivitäten auf einen Bruchteil der Akteure entfällt. Dies wurde als das 80/20-Pareto-Prinzip bekannt: 80 Prozent des Geschehens entfallen auf 20 Prozent der Beteiligten.

⁵⁶ Zehnder, a.a.O., S. 39, ebenso auch bei Fisher, *Myths and methods: a guide to software productivity*, New York, 1991, S. 35ff.

anfallenden und per EDV abzuwickelnden Vorgänge erledigen will, so steigen die Kosten nach dieser Annahme deutlich überproportional an. Einfacher ausgedrückt: Die Behandlung von Sonderfällen wird teuer.

c) Zeitpunkt und Zuständigkeit

Die Frage, zu welchem Zeitpunkt ein Pflichtenheft vorliegen soll, lässt sich aus Sicht des Software-Engineering nicht eindeutig beantworten. Dies liegt zum Teil an der dargestellten unterschiedlichen Terminologie oder der Uneinheitlichkeit in der Aufeinanderfolge der Arbeitsschritte im jeweiligen Modell. Teilweise scheint die Frage sich auch nicht zu stellen oder einer klaren Zuordnung zu entziehen⁵⁷. Dort, wo sie Berücksichtigung findet, fallen die Antworten insoweit einheitlich aus, als der Zeitpunkt jedenfalls nicht in einer frühen Phase des Projektes liegt⁵⁸. Ihm vorgelagert ist die Ist-Analyse und die grobe Festlegung des Sollzustandes⁵⁹. Konzeptionell (und logisch) nachgelagert ist jedenfalls die Phase der Implementierung. Das Pflichtenheft wird also regelmäßig im Stadium zwischen Sollkonzept und Implementation erstellt. Der genaue Zeitpunkt scheint maßgeblich davon abzuhängen, wieviel Unwägbarkeit die Vertragspartner zum Zeitpunkt der Festlegung in Kauf zu nehmen und wie flexibel sie mit allfälligen Änderungen⁶⁰ umzugehen bereit sind.

Hinsichtlich der Zuständigkeit für die Erstellung des Pflichtenheftes stellt sich die in der Rechtswissenschaft geführte Diskussion über die Abgrenzung von Verantwortlichkeiten des Softwarehauses und des Softwarebestellers nicht mit vergleichbarer Schärfe. Da der Softwarebesteller seine Bedürfnisse am besten kennt und andererseits das Softwarehaus bei der informationstechnischen Umsetzung einen Wissensvorsprung besitzt, ist der Planungsprozess „durch Kooperation bestimmt“⁶¹. Die Einbeziehung des

⁵⁷ So z.B. im ansonsten denkbar ausführlichen V-Modell.

⁵⁸ Franz, a.a.O., S. 65, 72, der ausdrücklich mehrere mögliche Zeitpunkte benennt, S.141 , Maymon/Cave, a.a.O., S. 50, 85; Love, a.a.O., S. 120; Suhl/Blumstengel, in: Fischer/Herold/Danglmayer/Nastansky/Suhl, a.a.O., S. 333.

⁵⁹ Suhl/Blumstengel, in: Fischer/Herold/Danglmayer/Nastansky/Suhl, a.a.O., S. 333; Franz, a.a.O., S. 65; Maymon/Cave, a.a.O., S. 85, die allerdings zumindest einen Teil der Istanalyse erst im Anschluss an die Anforderungsdefinition vorsehen, S. 50f; nicht eindeutig insoweit Love, a.a.O., S. 125.

⁶⁰ Siehe dazu sogleich unter II Änderungen.

⁶¹ Heinrich, a.a.O., S. 43.

Softwarebestellers als zukünftiger Benutzer wird als wichtiger oder sogar unverzichtbarer Aspekt der Entwicklung eingeschätzt⁶².

d) Voraussetzung für die Erstellung

Zur Erstellung eines verwertbaren Pflichtenheftes müssen zahlreiche Parameter einschließlich ihrer Gewichtung, gegenseitigen Abhängigkeit, Kosten usw. berücksichtigt werden⁶³. Als Hindernisse auf dem Weg werden eine Vielzahl von Faktoren genannt. Neben unklaren Zielvorstellungen⁶⁴ und Sprachbarrieren⁶⁵ gehören dazu ständig veränderte Ziele bzw. Anforderungen sowie das sog. Gold-plating⁶⁶.

Unklare Vorstellungen können leicht aus nicht funktionalen Anforderungen erwachsen. Dabei handelt es sich um Anforderungen, die nicht die Funktionalität betreffen, wie z.B. „einfache Bedienbarkeit“, „übersichtliche Gestaltung“⁶⁷. Sie müssen als Ziele des Softwarebestellers berücksichtigt werden, sollten dann aber möglichst quantifizierbar und damit einer Testbarkeit (und Abnahme) zugänglich sein⁶⁸. Insgesamt wird bei der Formulierung von Anforderungen der sprachlichen Präzision eine besondere Bedeutung beigemessen⁶⁹. Im Ergebnis muss der Softwarebesteller

- a) wissen, was er will und dies anschließend
- b) gemeinsam mit dem Softwarehaus zutreffend und präzise formulieren⁷⁰.

⁶² Etzel /Heilmann/Richter sprechen von konsequenter und kompromissloser Benutzerorientierung (a.a.O., S. 85); vgl. auch Fisher, a.a.O., S. 16; Zehnder, a.a.O., S. 190f; Etzel/Heilmann/Richter, a.a.O., S. 66.

⁶³ Dumke, a.a.O., S. 23ff.

⁶⁴ Rupp, a.a.O., S. 19.

⁶⁵ Rupp, a.a.O., S. 20.

⁶⁶ Darunter sind unnötige Funktionen zu verstehen (vergleichbar etwa einer Vergoldung des Schraubenziehers) mit entsprechenden Folgen für Aufwand und Kosten; Rupp, a.a.O., S. 22, ebenso Fisher, a.a.O., S.16.

⁶⁷ Auch hier ist die Parallele zu unbestimmten Rechtsbegriffen in der juristischen Terminologie augenfällig.

⁶⁸ Rupp, a.a.O., S. 264ff, 270.

⁶⁹ Die durch das Einhalten bestimmter Regeln zur Vermeidung sprachlicher Defekte erreicht werden kann, so Rupp (a.a.O.) mit instruktiver Übersicht auf S. 191ff. Dabei kann auch auf Schablonen (S. 230ff) und computergestützte Formulierungshilfen (S. 255) zurückgegriffen werden.

⁷⁰ Rupp, a.a.O., S. 191; Kratzsch, bei Rupp a.a.O. S. 421ff empfiehlt, die Anforderungen systematisch in einer Datenbank (und nicht einem einfachen Textdokument) zu erfassen, da sich so Fragen, Versionspflege und Abnahmekriterien leichter mit der jeweiligen Anforderung verknüpfen lassen.

Dieser scheinbar so einfach vorgezeichnete Weg ist nicht zuletzt infolge der Komplexität des zu beschreibenden Ergebnisses (der Software)⁷¹ nicht so einfach zu verfolgen, wie angenommen werden könnte⁷².

e) Anforderungen und Abnahme

Als Ziel der Entwicklung des V-Modells wurde die Verbesserung der Softwarequalität angegeben⁷³. Was ist nun unter Softwarequalität zu verstehen? Spätestens bei der Abnahme wird der Softwarebesteller zum Ausdruck bringen, ob das Gelieferte seinen Vorstellungen vom Vertragsgegenstand entspricht. Seine Einschätzung wird jedenfalls auch von subjektiven Vorstellungen bestimmt sein. Formuliert man mit Maymon, dass der entscheidende Maßstab für einen Projekterfolg die Anwenderzufriedenheit ist⁷⁴, so wird dieses subjektive Element deutlich. Aus diesem Grund wird die klare Definition davon nahegelegt, was unter Qualität zu verstehen ist⁷⁵. Zu den Qualitätskriterien werden neben der Funktionserfüllung und Benutzerfreundlichkeit die Kriterien Zuverlässigkeit, Wartungsfreundlichkeit⁷⁶ und Portabilität⁷⁷ genannt⁷⁸. Insbesondere bei den letztgenannten Kriterien handelt es sich um Merkmale, die nicht innerhalb kurzer Zeit überprüfbar und daher auch einer Abnahme nur schwer zugänglich sind. Dieser Problematik wird im Rahmen des Requirements Engineering dadurch Rechnung zu tragen versucht, dass Anforderungen und Abnahmekriterien miteinander verknüpft werden⁷⁹.

⁷¹ Nach Love sind größere (Software-) Projekte „so komplex in ihrem Bau und so teuer wie die Konstruktion eines großen Fabrikgebäudes oder neuen Flughafens“. a.a.O., S. 190.

⁷² Siehe dazu unten: Kritische Faktoren.

⁷³ S.o. S. 16.

⁷⁴ Maymon/Cave, a.a.O. S. 16, diesen subjektiven Faktor betonend auch Love, a.a.O., S. 206.

⁷⁵ Abzurufen ist daher nach Etzel/Heilmann/Richter von Vorgaben wie „weniger Benutzerfehler als bisher“. Stattdessen sollte konkret angegeben werden, z.B. wie viele Fehler auf wie viele Geschäftsvorfälle und wie viele Sekunden durchschnittlicher Antwortzeiten, a.a.O. S. 22ff.

⁷⁶ Hierunter wird v.a. die Änderbarkeit verstanden, siehe bei Franz, a.a.O., S. 19f.

⁷⁷ Darunter ist v.a. die Unabhängigkeit der Software von Rahmenbedingungen wie anderer Software, v.a. des Betriebssystems und der Hardware, zu verstehen, Franz, a.a.O., S. 21ff. Geht man davon aus, dass Individualsoftware erheblich länger „lebt“ als ihre technische Umgebung (so Franz, a.a.O., S. 85, Zehnder, a.a.O., S. 153f) so erlangt dieses Merkmal langfristig erhebliche Bedeutung.

⁷⁸ Etzel/Heilmann/Richter, a.a.O., S. 22; ausführlich Franz, a.a.O., S. 9ff.

⁷⁹ Nach Rupp sollten die Abnahmekriterien als Bestandteil des Vertrages innerhalb der Anforderungsanalyse erstellt werden (a.a.O., S. 328) und die Anforderungen präzise abbilden (S. 328).

Auf diese Weise soll erreicht werden, dass Softwarebesteller und Softwarehaus bereits zum Zeitpunkt der Pflichtenhefterstellung festlegen, woran der Projekterfolg gemessen werden wird. Allein die Tatsache, dass der Notwendigkeit dieses Zusammenhanges besondere Bedeutung beigemessen wird, wirft ein bezeichnendes Licht auf den Entwicklungsprozess von Software. Das Modell für die Softwareentwicklung (hier i.S. einer Skizze des fertigen Produktes) ist keine sinnlich wahrnehmbare Konstruktion wie das Modell eines Bürogebäudes oder einer Brücke, sondern ein auf Beschreibung in Worten und Zahlen angewiesenes Dokument.

2. Mitarbeit des Softwarebestellers

Wie oben⁸⁰ bereits angedeutet, ist die Rolle des Softwarebestellers auch und gerade im Rahmen der Erstellung des Pflichtenheftes von zentraler Bedeutung⁸¹. Die Gestaltung der Benutzeroberfläche wird sinnvollerweise vom Softwarebesteller bestimmt⁸². Seine Mitarbeiter sind es, die in den folgenden Jahren mit den entsprechenden Masken arbeiten werden. Geht man davon aus, dass ein Softwaresystem in erster Linie durch die Ausgaben, die es seinem Benutzer zur Verfügung stellt, definiert ist⁸³, so wird dieser Aspekt nochmals unterstrichen.

Hinsichtlich der richtigen Abbildung der Realität durch die Software, d.h. der vollständigen Auswahl der richtigen und wichtigen Daten, ist der Softwarebesteller zudem jedenfalls kompetenter als das Softwarehaus⁸⁴. Für das Einbinden der Software in die betrieblichen Abläufe und wenn es darum geht, die fachlichen (im Gegensatz zu den informationstechnischen) Anforderungen zu bestimmen, sind Führungskräfte aus dem Personal des Softwarebestellers unverzichtbar⁸⁵. Eschbach bemerkt demgemäß treffend,

⁸⁰ S.o. S. 21.

⁸¹ Schwickert betont für den Bereich des Web Site Engineering, also der Softwareentwicklung mit dem Ziel der (Neu-) Gestaltung einer Internetpräsenz, dass der Softwarebesteller nicht nur die strategischen Entscheidungen selbst vorbereiten und treffen sollte, sondern auch die Analyse und Modellierung der betriebswirtschaftlich-fachlichen Anforderungen von ihm wahrgenommen werden sollte (a.a.O., S 315f).

⁸² Eschbach, a.a.O., S. 125.

⁸³ So ausdrücklich Maymon/Cave, a.a.O., S. 84.

⁸⁴ Eschbach, a.a.O., S. 125.

⁸⁵ Etzel/Heilmann/Richter, a.a.O., S. 66, für eine optimale Teamzusammensetzung empfehlen die Autoren einen internen (d.h. aus dem Personal des Softwarebestellers stammenden) Abteilungsleiter und 9 weitere interne Mitarbeiter (7 aus der Informatik und zwei aus dem Fachbereich) gegenüber einem Projektleiter und 10 Softwareentwicklern

es sei ein gefährlicher Denkfehler, wenn man als Softwarebesteller meint, die eigene Beteiligung darauf beschränken zu können, Forderungen aufzustellen und auf die Einhaltung von Terminen zu drängen⁸⁶.

3. Änderungen

Ob es im Projektverlauf zu Änderungen der Anforderungen kommt oder nicht, stellt sich nicht als Frage dar, sondern wird als selbstverständliche Erfahrung aus der Praxis wiedergegeben⁸⁷. Den Grund für das regelmäßige Änderungsverlangen des Softwarebestellers sieht Brooks in der Unsichtbarkeit und Fügsamkeit des Produktes⁸⁸. Das Berücksichtigen dieser Änderungen kann auf verschiedene Weise geschehen. Zum einen sollte von Anfang an damit gerechnet werden, dass bereits erfolgte Entwicklungen wieder verworfen werden⁸⁹. Zum anderen kann dem Softwarebesteller schon zu einem frühen Zeitpunkt ein stark vereinfachter Entwurf vorgelegt werden. Diese „Prototyping“ genannte Methode wird deswegen verwendet, weil sie es dem Anwender am ehesten erlaubt, seine eigenen Vorstellungen anhand des Vorgefundenen zu artikulieren⁹⁰. Gleichwohl kann auch in diesem Fall nicht mit ihrem dauerhaften Bestand gerechnet werden⁹¹, aber die Methode der Leistungsbestimmung vermittelt zumindest Erfahrung und hilft frühzeitig dabei, Entwicklungen in die falsche Richtung zu korrigieren.

des Softwarehauses; Zehnder empfiehlt die Zusammensetzung der Projektleitung aus zwei Personen, wobei einer vom Softwarebesteller stammt (a.a.O., S. 188f), ferner soll ein qualifizierter Anwender zwingend in Vollzeit über einen gewissen Zeitraum im Projektteam mitarbeiten, um die Gestaltung der künftigen Funktionen und Arbeitsplätze zu ermöglichen (S. 190f).

⁸⁶ A.a.O., S. 75.

⁸⁷ Suhl/Blumstengel, in: Fischer/Herold/Danglmayer/Nastansky/Suhl, a.a.O., S. 372; Etzel/Heilmann/Richter, a.a.O., S. 174; Heinrich, a.a.O., S. 58; Rupp, a.a.O., S. 20f; nach Maymon/Cave ist es danach erforderlich, gut kontrollierte Änderungen bei den Benutzeranforderungen einzukalkulieren, a.a.O., S. 32.

⁸⁸ Er beschreibt dies als Gegensatz zu Hardware wie Computern oder Autos, Mythos, S. 102.

⁸⁹ Nach Brooks zeigt alle Erfahrung mit großen Projekten, dass man das erste Ergebnis nicht gebrauchen kann und es von Anfang an für den Abfalleimer bestimmen sollte, Mythos S. 102; Love spricht sogar davon, dass regelmäßig zweimal damit zu rechnen ist, Entwicklungen wieder wegzwerfen, wobei Wiederholungen den kompletten Entwicklungszyklus von Anforderungsanalyse bis Kundenabnahme umfassen, a.a.O., S. 239.

⁹⁰ „An existing system does more to clarify needs than any number of abstract definitions“, Fisher, a.a.O., S. 40; ebenso Etzel/Heilmann/Richter, a.a.O., S. 68; vgl. auch Suhl/Blumstengel, die im Oberflächenentwurf eine sehr gute Unterstützung für die Entwicklung sieht, aus der jedoch auch häufig Änderungen im Sollkonzept resultieren (in: Fischer/Herold/Danglmayer/Nastansky/Suhl, a.a.O., S. 377).

⁹¹ Eschbach geht davon aus, dass Prototypen i.d.R. weggeworfen werden und daher auch niemals als Grundlage für das eigentliche Programm dienen sollten (a.a.O., S. 128ff).

Dies ist im hier interessierenden Zusammenhang der Leistungsbestimmung auch deswegen besonders wichtig, weil konzeptionelle Fehler, d.h. solche, die auf fehlerhafter Analyse der Anforderungen beruhen, mit Tests erst spät entdeckt werden und daher exponentiell ansteigende Fehlerbehebungskosten zur Folge haben⁹². Ein Fehler in der Formulierung der Anforderungen ist daher deutlich teurer als ein Fehler, der bei der (falschen) Implementation der (richtigen) Anforderung auftritt.

Aus dem bisherigen Befund wird die Schlussfolgerung gezogen, dass

- (1) Änderungen im System vorzusehen sind⁹³,
- (2) bei grundlegenden Änderungen mit einem Neubeginn des Projektes zu rechnen ist⁹⁴,
- (3) notwendige Änderungen möglichst frühzeitig erkannt und berücksichtigt werden sollten,
- (4) Änderungsverlangen des Softwarebestellers, die dem Entwicklungsprozess zuwiderlaufen, Widerstand entgegenzusetzen ist⁹⁵.

a) Psychologie des Entwicklungsprozesses

Die Rolle des Softwarebestellers bei der Erarbeitung des Pflichtenheftes sowie die relative Häufigkeit von Änderungen lassen vermuten, was sich in der Praxis als essentiell für den Entwicklungsprozess erwiesen hat: Kommunikation⁹⁶. So sieht Eschbach in der Kommunikation zwischen Softwarehaus und Softwarebesteller den Schlüssel zum Projekterfolg⁹⁷. Dies gilt nicht nur hinsichtlich der beteiligten Projektmitarbeiter aus der EDV-

⁹² Nach einer Annahme von Barry Boehm verhalten sich die Kosten der Fehlerbehebung im Vergleich von Analyse:Design:Implementierung in der Größenordnung 1:10:100, zitiert nach Biedermann, Sven, in Rupp, a.a.O., S. 339f, von Biedermann mit dem Hinweis versehen, diese Aussage sei so wichtig für die Systementwicklung wie die Newton'schen Gesetze in der Physik; die Fehler der frühen (Planungs-) Phase sind also die bei weitem teuersten, Frühauf/Ludewig/Sandmayr, a.a.O., S.19; Franz, a.a.O., S. 128.

⁹³ Wie beispielsweise im V-Modell im Rahmen des Konfigurationsmanagements.

⁹⁴ Was nach Brooks bei mit dem Grundkonzept inkompatiblen Ideen angezeigt ist, Mythos, S. 39.

⁹⁵ Brooks spricht davon, dass Ideen und Vorstellungen, die nicht in das Gesamtkonzept passen, nur aus diesem Grund sofort verworfen werden sollten, Mythos, S. 39 dementsprechend bezeichnet er die „Geschlossenheit des Konzeptes“ als den (Hervorhebung im Original) grundlegenden Gedanken der Systementwicklung (S. 36).

⁹⁶ Brooks, der dies am Beispiel des Turmbaus zu Babel illustriert, Mythos, S. 64ff; Love, a.a.O., S. 120; Etzel/Heilmann/Richter, a.a.O., S. 64, 68.

⁹⁷ A.a.O., S. 105ff.

Abteilung des Softwarebestellers, sondern auch und gerade für dessen Fachabteilung und Führungskräfte⁹⁸.

Auch die Notwendigkeit klarer Verantwortungen und Zuständigkeiten⁹⁹ ist bei der Vielzahl beteiligter Personen unmittelbar einsichtig. Dies erstreckt sich auch auf die im Licht der Diskussion aktueller Entwicklungsmethoden vielleicht überraschende Forderung nach hierarchischen Strukturen¹⁰⁰, wobei allerdings Verantwortung und Entscheidungsbefugnis zusammenfallen müssen¹⁰¹.

Die bisher dargestellten Methoden und Erfahrungen dürfen nicht zu der Annahme verleiten, dass nur die strenge Beachtung ausgefeilter Vorgaben und Entwicklungssysteme zum Ziel führen kann¹⁰². Es gibt nicht wenige bekannte Beispiele für erfolgreiche Software (CP/M, MS-BASIC, MS-DOS, UNIX, Oracle, PASCAL, Lotus 1-2-3), die nicht unter Verwendung von Software-Engineering-Methoden und in sehr kleinen Teams entwickelt wurde¹⁰³. Dies mag auch an den erheblichen Produktivitätsunterschieden bei in der Entwicklung tätigen Personen liegen¹⁰⁴.

Hinsichtlich der Frage, wie umfangreiche Software rechtzeitig hergestellt werden kann, findet die ursprünglich von Brooks¹⁰⁵ stammende Aussage hinsichtlich des Zusammenhanges der Anzahl von Mitarbeitern und der Projektdauer allgemeine Unterstützung¹⁰⁶. Danach wird wegen des überproportional ansteigenden Kommunikationsbedarfes durch den Einsatz

⁹⁸ Etzel/Heilmann/Richter, der daneben auch das Einbeziehen des Betriebsrats empfiehlt, a.a.O., S. 64.

⁹⁹ Etzel/Heilmann/Richter, a.a.O., S. 76.

¹⁰⁰ Etzel/Heilmann/Richter, a.a.O., S. 78, Franz bemerkt dazu, dass eine Demokratie bei der Softwareentwicklung unmöglich sei, es müsse vielmehr eine Hierarchie geben (a.a.O., S. 31).

¹⁰¹ Nach Etzel/Heilmann/Richter das Kongruenzgesetz der Organisation, a.a.O., S. 26, d.h. werden projektbezogene Entscheidungen von Personen außerhalb des Projektes (beispielsweise einem Lenkungsausschuss) getroffen, so müssen diese auch für die sich daraus ergebenden Konsequenzen Rede und Antwort stehen.

¹⁰² Wie es z.B. Vorgehensweisen wie das V-Modell nahe legen könnten, andererseits schränken Vorgaben die Kreativität nicht zwangsläufig ein, sie können für einen kreativen Prozess vielmehr begünstigend wirken, ebenso Eschbach, a.a.O., S. 93.

¹⁰³ Love, a.a.O., S. 28 und bestätigend Etzel/Heilmann/Richter, a.a.O., S. 86, das bekannteste aktuelle Beispiel dürfte LINUX sein, dessen Kern bekanntlich aus der „Feder“ einer Person stammt. Es handelt sich allerdings durchweg um Standardsoftware.

¹⁰⁴ Love gibt an, die Produktivitätsabweichung von Softwareingenieuren könne 25:1 oder sogar 50:1 bei gleicher Ausbildung und Erfahrung (!) betragen (a.a.O., S. 244), auch arbeiten gute Teams laut Love bis zu 10 mal schneller als schlecht funktionierende (S. 248).

¹⁰⁵ Mythos, S. 22 „*der Einsatz zusätzlicher Arbeitskräfte bei bereits verzögerten Softwareprojekten verzögert sie nur noch mehr*“.

¹⁰⁶ Franz, a.a.O., S. 31; Zehnder, a.a.O., S. 180; Etzel/Heilmann/Richter, a.a.O., S. 24, Eschbach, a.a.O., S. 150.

zusätzlicher Arbeitskräfte keine Zeitersparnis erreicht¹⁰⁷. Auch der Gedanke, möglichst frühzeitig mit der „eigentlichen“ Programmierung zu beginnen¹⁰⁸, hat sich als Fehleinschätzung erwiesen¹⁰⁹. Sieht man in der Erstellung von Software nicht ohnehin eher eine Kunst als eine Wissenschaft¹¹⁰, so scheint nach dem Vorstehenden eine sorgfältige, vertrauensvolle und geduldige Erfassung und Fortschreibung des Projektzieles in Zusammenarbeit mit dem Softwarebesteller Voraussetzung für den Projekterfolg zu sein.

b) Kritische Faktoren in der Softwareentwicklung

Ein Grund für die vielgestaltigen Versuche, den Softwareentwicklungsprozess zu verstehen, zu systematisieren und für die Praxis tragfähige Konzepte zu entwerfen, hängt mit den häufigen Schwierigkeiten bis hin zum Scheitern von Projekten zusammen¹¹¹. Nach Franz werden 40 – 60 % aller Softwareprojekte nicht beendet¹¹². Nach Norvis halten lediglich 1 % mit hohem Softwareanteil das Budget und die Fristen und 25 % lieferten nie ein funktionierendes Produkt ab¹¹³.

¹⁰⁷ Dass auch Druck auf die Mitarbeiter einen rascheren Erfolg nicht unbedingt begünstigt, wird prägnant von Etzel/Heilmann/Richter zusammengefasst: Menschen denken unter Druck weder schneller noch besser, a.a.O., S. 24, ähnlich: Brooks mit zahlreichen Beispielen, Mythos, S. 26.

¹⁰⁸ von Brooks anschaulich als WHISKEY-Prinzip bezeichnet: „**W**hy in **H**eaven Isn't **S**omeone **K**oding **E**verything **Y**et ?“

¹⁰⁹ Eschbach formuliert: „*Je eher Du anfängst zu programmieren, desto länger wirst Du brauchen*“, a.a.O., S. 87, vgl. auch S. 114; Verschiedene Gründe dafür im Einzelnen auch bei Dumke, a.a.O., S.63.

¹¹⁰ So ausdrücklich Maymon/Cave, a.a.O., S. 173, wer die zahllosen Skizzen von Leonardo da Vinci, Dürer oder vieler anderer vor Augen hat, wird zwischen Wissenschaft und Kunst freilich keinen Widerspruch finden müssen.

¹¹¹ Love (a.a.O.) mit Beispielen S. 32.

¹¹² A.a.O. S. 30; nach Sneed, Risiken der Softwareentwicklung, zitiert nach Etzel/Heilmann/Richter (a.a.O., S.14) liegt die Wahrscheinlichkeit für den erfolgreichen Abschluss großer IT-Projekte (mit mehr als 20 Mitarbeitern) bei 42 %, bei Projekten mit weniger als 5 Mitarbeitern bei 74 %.

¹¹³ Mark Norvis, *Survival in the Software Jungle*, Boston, 1995 – zitiert nach Thaller, a.a.O., S. 25; auch wenn sich die statistischen Aussagen zahlenmäßig unterscheiden, so ist der Befund doch eindeutig: Wenn dasselbe für Bauvorhaben gelten würde, so stünde an jeder Straßenecke eine Bauruine.

Die Frage, ob und wann ein Projekt erfolgreich beendet wird, ist dabei auch deswegen schwer zu beantworten, weil auf Seiten der Informatik soweit ersichtlich die einheitliche Auffassung besteht, dass fehlerfreie Software nicht nachweisbar ist¹¹⁴. In der Praxis ist es weniger von Bedeutung, *ob* Fehler gefunden werden, sondern *wann* dies geschieht und wie Softwarebesteller und Softwarehaus darauf reagieren¹¹⁵.

Als eine wesentliche, wenn nicht sogar die bedeutendste Ursache für den Misserfolg von Projekten werden Schwierigkeiten bei den Anforderungsdefinitionen genannt¹¹⁶. Vyssotzky sieht die grundsätzliche Aufgabe der Systemarchitekten in einer Definition des Produktes, da viele Misserfolge ihre Ursache in den Punkten haben, die nie richtig spezifiziert worden waren¹¹⁷. Das Maß an Umsicht und Sorgfalt beim Erfassen der Anforderungen des Softwarebestellers erweist sich somit als zentrale Grundlage für den Projekterfolg¹¹⁸. Für das Ergebnis einzelner Projektabschnitte ist die Folge nachlässig formulierter Ziele anschaulich bei Brooks¹¹⁹ formuliert: „Der schwammig formulierte Meilenstein ist eine Bürde, an der man schwer zu tragen hat. Er ist im wahrsten Sinne des Wortes ein Mühlstein, der langsam die Moral des Teams zermalmt, weil er es über eventuelle Verzögerungen im Unklaren lässt, bis nichts mehr zu retten ist.“

¹¹⁴ Nach Zehnder ist aus Gründen der Komplexität normalerweise ein mathematischer Beweis für die „Richtigkeit“ des Gesamtsystems nicht möglich (a.a.O., S. 119); Ebenso Eschbach, der den vollständigen Test komplexer Software für unmöglich hält (a.a.O., S. 208).

¹¹⁵ Nach Maymon/Cave stellt sich häufig im Verlauf der Abschlusstests heraus, dass die Software nicht in dem Maße wie angenommen fertig war (a.a.O., S. 21), Software „*stirbt nicht. Sie wächst ständig.*“ (S. 7); ebenso Etzel/Heilmann/Richter, a.a.O., S. 75, 83); Eschbach führt dazu aus: Wenn die Programmierer das Gefühl haben, das Programm sei zu 90% fertig, dann ist „*Halbzeit*“. (a.a.O., S. 152).

¹¹⁶ Bei Barry Boehm, zitiert nach Thaller, a.a.O., S. 26, sind unvollständige Anforderungen mit 13,1 % der am häufigsten genannte Grund für den Abbruch von Software-Projekten; nach Rupp beruhen 65 % der schwerwiegenden Fehler in Programmen auf Unzulänglichkeiten in der (Anforderungs-) Analyse (a.a.O., S. 12); nach Zehnder ist der „Standardfehler“ in der Konzeptphase eine zu breite oder zu schmale Planung, d.h. es werden zu viele Details erfasst oder es wird nur ein Weg als der richtige angenommen (a.a.O., S. 59f); Etzel/Heilmann/Richter erläutern anhand eines instruktiven Beispiels, wie Missverständnisse bei der Bestimmung der Leistung zu einer hohen Gefährdung des Projekterfolges führen können (a.a.O., S. 29ff), vgl. auch Eschbach, a.a.O., S. 154.

¹¹⁷ V.A. Vyssotzky, Commonsense in designing testable software, nach Brooks, Mythos, S. 124.

¹¹⁸ „*Unsuccessful software projects fail in most cases because the software does not fit adequately to the needs of the people using it.*“, so Fisher, a.a.O., in seinem Vorwort S. IX; Ebenso deutlich auch Brooks: „*Eine sorgfältige Definition der Funktion und eine ebenso sorgfältige Spezifikation sind die Grundlage des Erfolgs.*“ Mythos, S. 124, vgl. auch Rupp, a.a.O., S. 25.

¹¹⁹ Mythos, S. 137.

Die Option eines Projektabbruchs, also die Frage, ob das Projekt fortgesetzt werden soll, ist in diesem Zusammenhang immer wieder zu stellen, weil der Abbruch sich als die bessere Alternative erweisen kann¹²⁰.

Neben der ungenauen Zieldefinition wird der beständige Wandel der Anforderungen als kritischer Faktor hervorgehoben. Dies gilt insbesondere, weil einmalige Abweichungen selten sind. Sie ziehen vielmehr regelmäßig weitere Änderungen nach sich¹²¹. Die Kosten für Änderungen (einschließlich Erweiterungen) können dabei in derselben Größenordnung wie die einer Neuentwicklung liegen¹²². Es wird daraus unmittelbar verständlich, warum die Erfolgsaussichten eines Projektes teilweise davon abhängig gemacht werden, wie wirkungsvoll das Softwarehaus sich gegen Änderungswünsche des Softwarebestellers wehren kann¹²³. Love¹²⁴ illustriert dies anschaulich, indem er als wichtigen Grund für ein Scheitern von Softwareprojekten in Bezug auf Erweiterungs- und Änderungswünsche ausführt: „Wir haben nicht gelernt, wie wir oft oder wirkungsvoll genug „nein“ sagen sollen.“

c) Zusammenfassung und Ergebnis

Die Erstellung von Software ist ein komplexer Prozess. Der Weg vom Beginn des Projektes bis zu seinem Abschluss ist in vielfältiger Weise begehbar. Die Vorstellung, für diesen Weg einen (allgemein-) verbindlichen Plan zu zeichnen, der die Wirklichkeit auch nur annähernd genau wiedergibt, hat sich in der Praxis als Illusion erwiesen¹²⁵. Andererseits wird genau dieser verbindliche und möglichst konkrete Plan als Grundvoraussetzung für ein Gelingen betrachtet. Softwarehaus und Softwarebesteller müssen also mit einem Paradoxon „zu Recht“ kommen: Es ist für einen Projekterfolg ebenso

¹²⁰ Zehnder, a.a.O., S. 144, ebenso Franz, a.a.O., S. 66.

¹²¹ Etzel/Heilmann/Richter, a.a.O., S. 24; Brooks gibt an, dass jede Instandsetzung eines Programmes sehr wahrscheinlich 20-50% neue Fehler nach sich zieht; Mythos, S. 107.

¹²² was nach Zehnder sogar oft der Fall ist, a.a.O., S. 174.

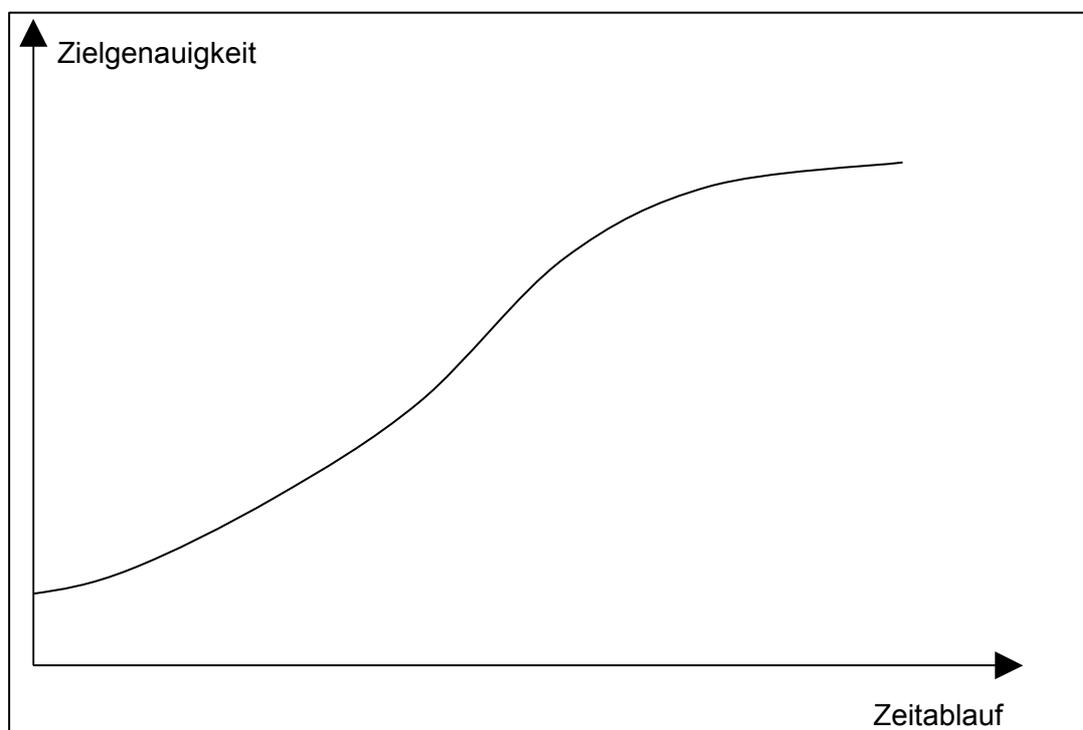
¹²³ Zehnder sieht die konsequente Bekämpfung von Projektänderungen als zentrale Forderung an eine gute Projektleitung (a.a.O., S. 294f); Brooks bemerkt, dass richtiges Projektmanagement danach verlangt, „rückhaltlos, plausibel (und manchmal unter dem Damoklesschwert des Auftragsverlustes) Planungen zu vereiteln, die jeglicher Grundlage entbehren!“ Mythos, S. 19.

¹²⁴ A.a.O., S. 33.

¹²⁵ Kargl formuliert dies mit der trockenen Definition: „Bei komplexen Projekten besteht eine niedrige Planbarkeit, ein im Voraus nicht bestimmbarer Informationsbedarf, ein hoher Bedarf an Kommunikation und vielfältige Möglichkeiten der Bearbeitungsfolge.“ (a.a.O., S. 110).

zwingend notwendig wie unmöglich, präzise ein Ziel vorzugeben. Zwingend notwendig ist es, weil nur auf diese Weise die Vorstellungen und Wünsche des Softwarebestellers, die das Projekt ins Leben gerufen haben, realisiert werden können. Unmöglich ist es, weil die Vorstellungen und Wünsche sich im Laufe des Projektes aufgrund sich ändernder Rahmenbedingungen oder unabhängig davon ebenfalls ändern¹²⁶. Der Änderungsbedarf kann sich auch aus Kenntnissen ergeben, die überhaupt erst im Projektverlauf gewonnen werden konnten.

Ein denkbarer Verlauf für die Zielgenauigkeit ist in der folgenden Grafik abgebildet.



Die Zielgenauigkeit wird hier mit Zeitablauf infolge des Informationszugewinns zunächst größer, das Maß des Zuwachses verlangsamt sich jedoch mit zunehmender Informationsdichte¹²⁷.

Der beste Zeitpunkt für eine (verbindliche) Zieldefinition liegt demnach eindeutig nicht bei Projektbeginn.

¹²⁶ Neben den hier behandelten "problematischen" Fällen gibt es auch Situationen, in denen das Ziel dem Softwarehaus von Beginn an sehr präzise vorgegeben werden kann. Dies ist etwa bei der Programmierung technisch definierter Abläufe der Fall. Der Softwarebesteller legt dem Softwarehaus die definierten Eingangs- und die definierten (durch das Softwarehaus programmtechnisch zu erarbeitenden) Ausgangsdaten vor.

¹²⁷ Es tritt ein "Überfütterungseffekt" ein: Die Vielzahl der hinzutretenden Informationen kann bei konstanten Verarbeitungsressourcen nicht bewältigt („verdaut“) werden. Dies kann u.U. auch zu einem rückläufigen Verlauf der Kurve führen, d.h. die Zielgenauigkeit nimmt nicht zu, sondern ab.

Die Verantwortlichkeit für die Zieldefinition kann eindeutig weder dem Softwarehaus noch dem Softwarebesteller zugeordnet werden. Ohne Kommunikation der Vertragspartner und kontinuierliche Mitarbeit des Softwarebestellers kann jedenfalls kein optimales Ergebnis erzielt werden.

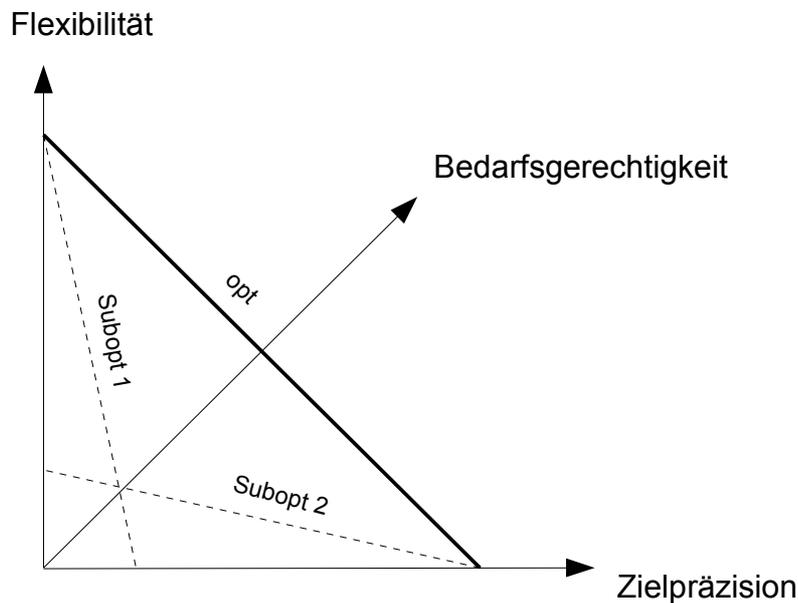
Ein Grund für Änderungen der Anforderungen sind Fehlvorstellungen über die Einfachheit und die Auswirkungen von Änderungen. Software ist ein hochgradig wandlungsfähiges und wandlungsanfälliges Produkt. Häufige Änderungen tragen jedoch nicht zum Projekterfolg bei, da Änderungen an einer Stelle regelmäßig die Notwendigkeit von Änderungen an anderer Stelle und damit einen nicht kalkulierbaren Ressourcenverbrauch (Zeit und Kosten) nach sich ziehen. Je später Änderungen konzeptioneller Art durchgeführt werden, um so aufwendiger sind sie.

Schließlich hat sich Fehlerfreiheit¹²⁸ nicht als geeignetes Qualitätskriterium für komplexe Software erwiesen.

Die Grundvoraussetzungen, unter denen ein Softwareprojekt durchgeführt wird, sind demnach im Anschluss an die Ausgangsfrage aus dem ersten Kapitel wie folgt zu benennen:

Eine effektive Projektabwicklung, d.h. die optimale Zielerreichung, hängt von stabilen und präzisen Vorgaben bei flexibler Handhabung ab. Präzise und unveränderte Zielvorgaben erhöhen die Kalkulierbarkeit, wobei sie jedoch bei mangelnder Flexibilität im Hinblick auf allfällige Änderungen zu einer unbrauchbaren Lösung führen können. Ein bedarfsgerechtes Ergebnis wird also nur erreicht, wenn die Genauigkeit der Spezifikation mit der Offenheit für Änderungen ausbalanciert ist. Dies kann schematisch in Form einer Grafik dargestellt werden, wobei das Maß, in dem ein Softwareprojekt bedarfsgerechte Lösungen hervorbringt, davon abhängt, ob es gelingt, Zielpräzision und Wandelbarkeit in eine optimale Balance zu bringen.

¹²⁸ Dies schließt nicht aus, dass in manchen Situationen eine Fehlertoleranz nicht akzeptabel ist (Wer würde sich sonst wohl der Softwaresteuerung eines Aufzuges oder einer Notversorgung (z.B. Sauerstoff- und Medikamentenzufuhr) anvertrauen ?!).



Wird die Zielpräzision bei geringer Flexibilität maximiert (Linie *subopt 2*), so führt dies zu einem vergleichsweise geringen Ausschlag auf der Achse der Bedarfsgerechtigkeit. Ebenso verhält es sich, wenn die Flexibilität gesteigert wird, die Zieldefinition aber auf einem geringen Niveau verbleibt (Linie *subopt 1*). Erst wenn Flexibilität und Zielpräzision in gleichem Umfang, d.h. ausbalanciert, gesteigert werden, wird die Aussicht auf eine bedarfsgerechte Lösung optimal unterstützt¹²⁹.

Schließlich kann als weiteres Ergebnis der Untersuchung festgestellt werden, dass der Projekterfolg von einer angemessenen Einbindung des Softwarebestellers in die Projektarbeit maßgeblich beeinflusst wird. Dies gilt vor allem hinsichtlich der Formulierung und Präzisierung der Anforderungen.

¹²⁹ Hier geht es lediglich um das Veranschaulichen prinzipieller Zusammenhänge schwerlich quantifizierbarer Merkmale. In der Praxis werden die Gegebenheiten, von Projekt zu Projekt verschieden, einen anderen Winkel für die Darstellung der Bedarfsgerechtigkeit erfordern, z.B. abhängig vom Verhältnis funktionaler und nicht funktionaler Anforderungen.

2. AMERIKANISCHER TEIL

Kapitel 3: Rechtliche Rahmenbedingungen für Softwareentwicklungsverträge

A. Einordnung und Bedeutung der Rechtsquellen

Das amerikanische Recht unterscheidet sich bereits in systematischer Hinsicht grundlegend vom deutschen. Zwei Besonderheiten sind dabei für den Untersuchungsgegenstand in erster Linie zu berücksichtigen: Das Verhältnis von Gesetz und Rechtsprechung und das Primat der einzelstaatlichen Regelung.

Im Verhältnis von Gesetz und Rechtsprechung dominiert im angloamerikanischen Rechtskreis die Entscheidung des Gerichts. Nicht das abstrakte Gesetz, sondern die im gerichtlichen Verfahren gewonnene Erkenntnis wird zur wegweisenden Regel. Die Gesetzgebung kann damit nur eingeschränkt als Ausgangspunkt oder Eckpfeiler für die rechtliche Beurteilung von Verträgen herangezogen werden.

„Codes“ sind in der Common Law-Rechtskultur als eine Hilfe, nicht jedoch als Grundlage der Rechtsfindung zu verstehen¹³⁰.

Die Bedeutung des Wortlauts der vertraglichen Vereinbarung sticht als individuelles Element demgegenüber deutlich hervor und drängt teleologische und historische Auslegung, insbesondere auch das Berücksichtigen vertragsbegleitender Umstände, in den Hintergrund. Dieses Prinzip steht im Gegensatz zu der deutschen Regelung des § 133 BGB, wonach der wirkliche Wille und nicht der buchstäbliche Sinn des Ausdrucks für eine Auslegung maßgeblich ist. Ohne Kenntnis dieses Prinzips ist die in

¹³⁰ „Where such codes (...wie der U.C.C.) exist, they make no pretense of completeness. The judge is not compelled to find a basis for deciding a given case within the case. Usually, moreover, such codes are not rejections of the past; they do not purport to abolish all prior law in their field, but rather to perfect it and, except where it conflicts with their specific present purposes, to supplement it. Where some provision of a code or other statute appears to be in possible conflict with a deeply rooted rule of Common Law, the tendency will be to interpret the code provision in such a way as to evade the conflict.“
Merryman, John Henry, *Western European and Latin American Legal Systems*, Charlottesville Virginia, 1978, S. 829.

amerikanischen Verträgen zu beobachtende Tendenz zu umfassender Regelung und die damit einhergehende Länge der Vertragstextes kaum nachvollziehbar.

Das Primat der einzelstaatlichen Regelung fußt auf der föderalen Struktur der Vereinigten Staaten. Anders als im kontinentaleuropäischen Rechtsraum gibt es keine einheitliche Kodifikation des Vertragsrechts. Nach der Kompetenzverteilung der US-amerikanischen Verfassung ist die Regelung des Vertragsrechts den Einzelstaaten vorbehalten¹³¹. Damit dies nicht zu 50 gänzlich verschiedenen Systemen in einem sprachlich, kulturell und wirtschaftlich verbundenem Wirtschaftsraum führt, existieren sog. Uniform Laws¹³². Diese haben allerdings nur Vorbildfunktion (default rule), d.h. sie können, müssen aber nicht von den Einzelstaaten umgesetzt werden¹³³.

Im amerikanischen Recht gibt es kein besonderes Schuldrecht i. S. des BGB¹³⁴, das versucht, alle Vertragsarten zu erfassen. Insbesondere spezielle gesetzliche Regelungen für die Herstellung eines Werkes (Werkvertragsrecht) sind nicht vorhanden¹³⁵. Im Rahmen des Untersuchungsgegenstandes muss daher auf allgemeinere Regelungen zurückgegriffen werden. Für das amerikanische Recht sind dabei vor allem der U.C.C.¹³⁶ und die Restatements of the Law of Contracts von besonderer

¹³¹ Das Amendment X. (aus 1791) zur amerikanischen Verfassung besagt: „*The powers not delegated to the United States by the Constitution, nor prohibited by it to the states, are reserved to the states respectively, or to the people.*“ Dies verkörpert ein den Art. 30, 70 GG innewohnendes Prinzip. Das bürgerliche Recht wird jedoch nicht wie in Art. 74 Abs. 1 Nr. 1 GG dem Bund zugeordnet.

¹³² Wie der Uniform Commercial Code, siehe unten S. 40.

¹³³ Der U.C.C. ist von allen Staaten und Territorien mit Ausnahme von Louisiana übernommen worden, was jedoch wegen selbständiger Interpretation und Umsetzungsvarianten nicht notwendig zu einer einheitlichen Rechtslage führt, vgl. Blumenwitz, Einführung in das anglo-amerikanische Recht, 7. Aufl., München, 2003, S. 48 Fn. 13.

¹³⁴ Für einen Common-Law-Juristen kann bereits die Zweckmäßigkeit der Unterscheidung von öffentlichem und privatem Recht fraglich oder zumindest nicht zwingend sein, Merryman, John Henry, *Western European and Latin American Legal Systems*, a.a.O., S. 800ff, die praktische Bedeutung dieser Betrachtungsweise zeigt sich z.B. im amerikanischen *unified judicial system*, das die dem Civil Law System geläufige Unterscheidung von Verwaltungs- und Ordentlicher Gerichtsbarkeit nicht kennt, Merryman, *The Civil Law Tradition*, Sec. ed., Stanford California 1985.

¹³⁵ Im Uniform Commercial Code sind u.a. Kauf (U.C.C. - ARTICLE 2 – SALES), Miete (U.C.C. - ARTICLE 2A – LEASES) und Darlehen (U.C.C. - ARTICLE 3 - NEGOTIABLE INSTRUMENTS) geregelt, nicht jedoch z.B. Dienst- und Werkvertrag.

¹³⁶ Der Uniform Commercial Code wurde der Öffentlichkeit durch die National Conference of Commissioners on Uniform State Laws erstmals 1952 vorgestellt, jedoch zunächst nur in Pennsylvania in geltendes Recht transformiert. Nach ausgiebigen Änderungen in den folgenden Jahren wurde der U.C.C. zwischen 1957 und 1967 in allen Staaten eingeführt. Nur Louisiana beschränkte sich auf die Einführung der Art. 1, 3, 4 und 5 (Corbin, Arthur

Bedeutung¹³⁷. Daneben tritt für zwischenstaatliche Verträge das UN-Kaufrecht¹³⁸, das als Federal Law das sonstige State Law (einschließlich des U.C.C.) verdrängt und für alle Gerichte im Rahmen seines Anwendungsbereiches¹³⁹ verbindlich ist¹⁴⁰.

Als speziellste Regelung ist schließlich der ebenfalls als default-rule auf den Weg gebrachte Uniform Computer Information Transactions Act (UCITA)¹⁴¹ zu nennen.

B. Performance

Nach einer dem Anspruchsaufbau des BGB entsprechenden Norm wie § 631 BGB, die eine konkrete dem Vertragstypus zugeordnete Pflicht benennt, sucht man im amerikanischen Recht vergeblich. Als primärer Rechtsschutz im Fall von Vertragsverletzung wird Schadensersatz (damages) angesehen¹⁴².

In der Regel wird daher Schadensersatz zugesprochen und nur in Ausnahmefällen auf eine Verpflichtung zur Erfüllung des Vertrages (*specific performance*) erkannt.

Specific performance wird demnach nur in einer begrenzten Auswahl von Situationen zugestanden¹⁴³. Herstellungsverträge (construction contracts)

Linton, Corbin on Contracts, Revised Edition, St. Paul, 1993, S. 76ff; Macneil, Ian R./Gudel, Paul J., Contracts, 3. Auflage, New York, 2001, S. 94ff). Auch an dem Entwurf und der Weiterentwicklung des U.C.C., v.a. im Wege der offiziellen Kommentierung ist das American Law Institute maßgeblich beteiligt (vgl. Macneil/Gudel, S. 95).

¹³⁷ Das angesehene American Law Institute (ALI) veröffentlichte sein Restatement 1932 zum ersten Mal. Die Konzeption ist der eines Gesetzes vergleichbar (385 §§ in drei Bänden, 16 Kapiteln, die in sog. Topics unterteilt sind – z.B. in dem Kapitel 3 „Formation of Contracts - Mutual Assent“ das Topic „In General“ gefolgt von „Manifestation of Assent in General“ und „Making of Offers“ usw. Die zweite Auflage der Restatement of the Law of Contracts wurde 1962 begonnen, 1979 (!) abgeschlossen und 1981 veröffentlicht. Ihre Bedeutung wird von Calamari/Perillo mit den Worten „*it is a highly persuasive authority*“ gekennzeichnet (The Law of Contracts, 4th Edition, St. Paul 1998, S. 15).

¹³⁸ Convention on the International Sale of Goods (CISG), prepared by the United Nations Commission on International Trade Law (UNCITRAL), and adopted by a diplomatic conference on 11 April 1980, adopted by the United States on December 11th 1986, effective on January 1, 1988.

¹³⁹ Zu der Frage, ob Software in den Anwendungsbereich der CISG fällt, s.u. S.47.

¹⁴⁰ Macneil/Gudel, a.a.O., S. 96.

¹⁴¹ Ursprünglich vorgesehen als Article 2B des Uniform Commercial Code (U.C.C. 2B). Der Originaltext ist erhältlich über die University of Pennsylvania Law School unter: <http://www.law.upenn.edu/blil/ulc/ulc.htm#ucita>, siehe unten unter III.

¹⁴² Calamari/Perillo, a.a.O., S. 611.

¹⁴³ Calamari/Perillo, a.a.O., S. 611ff, dazu zählen traditionsgemäß Immobiliengeschäfte (Real property) und in der neueren Rechtsprechung Dienstverträge (Service contracts, S. 614) mit Sportlern und Künstlern (S. 618).

gehören typischerweise nicht dazu¹⁴⁴. Erst wenn die geschuldete Leistung wegen ihrer Einzigartigkeit nicht durch Erfüllung von anderer Seite erbracht werden kann, besteht Aussicht auf Erzwingung im Wege gerichtlicher Entscheidung. Bei der Durchsetzung von *Schiedssprüchen*, welche specific performance zum Gegenstand haben, kommt es auf die Eigenart des Vertrages jedoch nicht an¹⁴⁵.

Die Bestimmtheit der vertraglichen Vereinbarung, zumindest in Form einer reasonable certainty, ist Voraussetzung für specific performance¹⁴⁶.

Als im Rahmen der Courts of Equity entwickeltes Rechtsinstitut baut specific performance auf dem Grundstein der unconscionability auf¹⁴⁷. Da die Anwendung und Auswirkung der unconscionability in hohem Maße von den besonderen Umständen des Einzelfalles abhängt, ergeben sich weite Ermessensspielräume für die zur Entscheidung berufenen Gerichte¹⁴⁸.

Bemerkenswert ist schließlich die Feststellung, specific performance sei generell inkonsistent mit dem Zusprechen von Schadensersatz¹⁴⁹.

Im Ergebnis wird ein Erfüllungsanspruch im amerikanischen Recht nur unter engen Voraussetzungen gewährt. Einen Teil dieser Voraussetzungen können die Vertragspartner im Wege präziser Beschreibung der geschuldeten Leistung sowie ggf. der einzigartigen Expertise eines Vertragspartners selbst schaffen.

Doch auch wenn eine Leistungsbestimmung erfolgt, so haben im Konfliktfall die Gerichte die grundsätzliche Frage zu klären, um welchen Vertragstypus es sich handelt, insbesondere ob goods oder services geschuldet sind. Im Ergebnis wird eine Durchführung des Vertrages über die Rechtsfigur der specific performance nur in Ausnahmefällen zu erzwingen sein.

¹⁴⁴ Als Begründung hierfür wird die Schwierigkeit der Überwachung einer Vertragsdurchsetzung und die Unangemessenheit des Erzwingens einer Fortsetzung unerwünschter persönlicher Beziehungen („*forcing the continuance of a distasteful personal relationship*“, Calamari/Perillo, a.a.O., S. 617f und 623) angegeben. Die Schwierigkeit der Überwachung wird jedoch in zunehmendem Maße nicht als überzeugendes Argument angesehen („*Indeed, the willingness of courts of equity in recent decades to take on supervision of complex school desegregation and legislative reapportionment plans, would indicate that supervision of contract performance is a burden that courts can deal with.*“ Calamari/Perillo, a.a.O., S. 623.

¹⁴⁵ Calamari/Perillo, a.a.O., S. 619.

¹⁴⁶ Calamari/Perillo, a.a.O., S. 622.

¹⁴⁷ Calamari/Perillo, a.a.O., S. 611 und S. 366

¹⁴⁸ So hieß es im Restatement 1st § 364, dass Billigung oder Zurückweisung der specific performance von „*the moral standards of enlightened judges*“ abhängt.

¹⁴⁹ Calamari/Perillo, a.a.O., 16.18 (S. 631).

C. Statute Law

I. U.C.C.

1. Anwendbarkeit

Der U.C.C. ersetzt im Rahmen seines Anwendungsbereiches das Common Law¹⁵⁰.

Die Bedeutung des U.C.C. erschöpft sich dabei nicht in der Regelung der von ihm ausdrücklich erfassten Materie. Das Statute Law muss infolge des oben angesprochenen Verhältnisses von Gesetz und Rechtsprechung als „weiches“ Recht verstanden werden. Die Gerichte sind einerseits an einer Rechtsfortbildung des angewendeten Rechts nicht gehindert, andererseits zu einer Anwendung der in unanwendbarem Recht verkörperten Prinzipien in der Lage.

So heißt es im offiziellen Kommentar zu U.C.C. § 1-103 (U.C.C. 2003 Edition, Philadelphia Chicago 2003, S. 23):

„Courts have recognized the policies embodied in an act as applicable in reason to subject-matter which was not expressly included in the language of the act (...). They have done the same where reason and policy so required, even where the subject matter has been intentionally excluded from the act in general (...). Nothing in this act stands in the way of continuance of such action by the courts.“¹⁵¹

Der U.C.C. wird von den Gerichten demgemäß häufig als Quelle für das Common Law herangezogen¹⁵², und zwar dem Zitat gemäß auch in Fällen, in denen eine direkte Anwendung ausgeschlossen ist.

¹⁵⁰ U.C.C. § 2-102.

¹⁵¹ Die Bedeutung des offiziellen Kommentars für die Auslegung des U.C.C. kann mit den Worten von Macneil/Gudel (a.a.O., S. 95) vor Augen geführt werden: „*Thus, while not a part of the statute, the Official Comments are extremely authoritative in interpreting the U.C.C.*“

¹⁵² Corbin, a.a.O., S. 79ff.

Einschränkend ist allerdings darauf hinzuweisen, dass der U.C.C. in seiner ursprünglichen, d.h. vom Permanent Editorial Board herausgegebenen, Form praktisch nicht existiert. Viele Einzelstaaten haben den Einheitstext an verschiedenen Punkten geändert. Zudem sieht der Einheitstext verschiedene optionale Regelungen vor, von denen zahlreiche Staaten ebenfalls Gebrauch gemacht haben¹⁵³. Dieser theoretische Befund darf jedoch nicht darüber hinwegtäuschen, dass neben anderen integrierenden Faktoren nicht zuletzt durch national (d.h. für den Bereich der gesamten Vereinigten Staaten von Amerika) ausgeprägte Ausbildung, Forschung und Prüfung in weiten Gebieten durchaus ein bundeseinheitliches Vertragsrecht gelehrt, gelernt, geprüft und damit praktiziert wird¹⁵⁴.

Der U.C.C. besteht in seiner aktuellen Fassung aus 9 Teilen (*Articles*), von denen vor allem der zweite (*Sales*) hier von Interesse ist¹⁵⁵. Art. 2 U.C.C. ist nur auf Kaufverträge (*Sales, Art. 2*) anwendbar. Es fragt sich daher zunächst, ob Software unter den Anwendungsbereich des U.C.C. fällt.

2. Anwendbarkeit des U.C.C. auf Software

Artikel 2 des U.C.C ist ausdrücklich nur auf „transactions in goods“ anwendbar¹⁵⁶. „Goods“ sind dabei definiert als „all things (including specially manufactured goods) which are movable at the time of identification to the contract for sale“¹⁵⁷. Computer hardware wird unzweifelhaft als „goods“ eingeordnet, während Verträge, die vorwiegend Dienstleistungen („personal services“) beinhalten, wie Wartungs-, Trainings- oder Supportverträge, häufig¹⁵⁸ als nicht vom U.C.C erfasst angesehen werden.

¹⁵³ Calamari/Perillo, a.a.O., S. 17.

¹⁵⁴ Dazu ausführlich Eisenberg, Melvin A., 50 States but one law ?; in: Weyers (Hrsg.), Europäisches Vertragsrecht, Baden-Baden, 1997, S. 30ff.

¹⁵⁵ Die aktuelle Fassung des Modellentwurfes findet sich unter <http://www.law.cornell.edu/ucc/ucc.table.html>, die in den Einzelstaaten umgesetzten Fassungen finden sich unter <http://www.law.cornell.edu/uniform/ucc.html>, die aktuellen Änderungsvorschläge für Art. 2 - U.C.C. sind dokumentiert bei Flechtner, Harry M., Substantial Revisions to US Domestic Sales Law, IHR, 2004, S. 225ff.

¹⁵⁶ U.C.C. § 2-102.

¹⁵⁷ U.C.C. § 2-105(1).

¹⁵⁸ Jedoch nicht ausschließlich: In einigen Fällen wurde der U.C.C. auch dann für einschlägig befunden, wenn in einem Dienstleistungsvertrag die Lieferung von „goods“ eingeschlossen war, wobei die Anwendbarkeit des U.C.C. mit dem Umfang der gelieferten „goods“ wahrscheinlicher ist, Scott, Scott on Computer Law, 2. Aufl., New York, 2003, § 7.09[A] S. 7-28.

Ist der Vertragsgegenstand Software, so erhebt sich die viel diskutierte Frage nach deren Sacheigenschaft (*movable thing*).

Bei den Entscheidungen, die eine Einordnung von Software als good – und damit die Anwendbarkeit des U.C.C. – bejahen, können zwei Gruppen von Argumenten unterschieden werden: Praktikabilitätsgesichtspunkte einerseits und funktionelle Überlegungen andererseits.

Zu den letzteren zählt zunächst die Überlegung, dass es sich auch bei einem Vertrag über Software um einen „contract to give“ handle, und die dominierende oder grundsätzliche Verpflichtung darin bestehe, etwas zu geben oder zu liefern¹⁵⁹. Darüber hinaus wird auf das Ergebnis, welches vom Programm geliefert wird, und nicht auf das Programm selbst abgestellt: Das Ergebnis sei gegenständlich und nur darauf komme es an¹⁶⁰. Die Sacheigenschaft von Software ergäbe sich zwangsläufig aus der Notwendigkeit, sie in ein Medium zu übertragen¹⁶¹. Durch das Trägermedium erhalte sie ein „tangible home“ und erst durch die Verkörperung gewinne sie einen Wert. Die Verbindung von Medium und Information sei daher unentwirrbar¹⁶². Auf die in das Produkt investierte Arbeit schließlich komme es nicht an, da Arbeit als Dienstleistung in jedem Produkt enthalten sei und daher auch jeder Verkauf von Gütern einen hybriden Vertrag darstelle¹⁶³. Zu den praktischen Gesichtspunkten gehört die Überlegung, dass Softwareverkäufe Alltagsgeschäfte darstellen und die Parteien berechtigterweise erwarten können, zumindest *ein* Gesetz regele den Verkauf, und der U.C.C. erfülle diese Erwartung eher als das Common Law¹⁶⁴. Immer wieder findet sich zudem die Auffassung, der

¹⁵⁹ Austin's of Monroe, Inc. v. Bill Brown, 474 So. 2D 1383; Supreme Judicial Court of Massachusetts; 4.02.80.

¹⁶⁰ Communications Groups, Inc., Also Known as CGI, Plaintiff, v. Warner Communications Inc., 527 N.Y.S.2d 341; Systems Design and Management Information, Inc., (SDMI) v. Kansas City Post Office Employees Credit Union, 788 P.2d 878, N. Distr. Court. Texas, 28.04.03.

¹⁶¹ Advent Systems Ltd. v. Unisys Corp., 925 F.2d 670.

¹⁶² Comshare, Inc., v. United States of America, 27 F.3d 1142, US Distr. Ct. S.D. N.Y., 03.09.86 in einem steuerrechtlichen Fall. Auf die Übertragung einer Idee in eine „tangible expression“ stellt auch ab: Management Computer Servs., Inc. v. Hawkins, Ash, Baptie & Co, 557 N.W.2d 67, US Cr. Appeals 6 Circuit, 27.06.94.

¹⁶³ Micro Data Base Systems, Inc. v. Dharma Systems, Inc., 148 F.3d 649, Appeal from the United States District Court for the Eastern District of Virginia, at Alexandria, 13.09.95; Populsion Technologies, Inc., d/b/a PowerTech Marine Propellers, v. Attwood Corporation, 369 F.3d 896, US District C. Northern Ohio East, 27.09.01.

¹⁶⁴ I.Lan Systems, Inc. v. Netscout Service Level Corp., 183 F.Supp.2d 328, District Cr. New Hampshire, 31.03.93.

Anwendungsbereich des U.C.C. sei großzügig auszulegen, um die ihm zugrunde liegenden Ideen zu fördern¹⁶⁵. Dafür spräche zudem, dass eine Vereinfachung und Einheitlichkeit der Rechtsanwendung nur auf diese Weise erreicht werde¹⁶⁶.

Für die Einordnung von Software als service wird im Gegensatz zu dem Vorstehenden vorgebracht, beim Erwerb von Software stehe das Wissen und die Fähigkeiten des Programmierers im Vordergrund. Diese und nicht die Gegenstände, mit deren Hilfe Wissen und Fähigkeiten in den Computer übertragen werden, seien die Essenz eines Softwareerwerbs¹⁶⁷. Es handele sich also um einen "contract to do"¹⁶⁸ bzw. einen "contract for work labour and materials and not a sale"¹⁶⁹. Der Charakter von Software als eine klassische Form geistigen Eigentums werde nicht dadurch beeinträchtigt, dass sie aus einer physischen Komponente und einer immateriellen -nicht berührbaren- Komponente bestehe.¹⁷⁰

3. Anwendbarkeit auf Softwareentwicklung

Bei Softwareentwicklungsverträgen liegt bei Vertragsabschluss, wie bei Lieferverträgen über vorher nicht existierende Sachen, noch keine fertige Software vor. Die Mehrzahl der amerikanischen Gerichte¹⁷¹ und der

¹⁶⁵ Communications Groups, Inc., Also Known as CGI, Plaintiff, v. Warner Communications Inc., 527 N.Y.S.2d 341; Systems Design and Management Information, Inc., (SDMI) v. Kansas City Post Office Employees Credit Union, 788 P.2d 878, N. Distr. Court. Texas, 28.04.03; Comshare, Inc., v. United States of America, 27 F.3d 1142, US Distr. Ct. S.D. N.Y., 03.09.86; Propulsion Technologies, Inc., d/b/a PowerTech Marine Propellers, v. Attwood Corporation, 369 F.3d 896, US District C. Northern Ohio East, 27.09.01.

¹⁶⁶ Systems Design and Management Information, Inc., (SDMI) v. Kansas City Post Office Employees Credit Union, 788 P.2d 878, N. Distr. Court. Texas, 28.04.03; Advent Systems Ltd. v. Unisys Corp., 925 F.2d 670.

¹⁶⁷ LeSueur Creamery, Inc. v. Haskon, Inc., 660 F.2d 342, US Court of Appeals 8 Circ., 17.9.81; Data Processing Services, Inc. v. L. H. Smith Oil Corp., 492 N.E.2d 314, Ct. App. Indiana 4 Distr., 28.4.86.

¹⁶⁸ Austin's of Monroe, Inc. v. Bill Brown, 474 So. 2D 1383, Ct. Appeal Louisiana 2d Circuit, 21.8.85.

¹⁶⁹ Micro Managers Inc. v. Stanley Oren/Terry Coleman, 147 Wisc. 2D 500, Ct. Appeals Wisc., 10.11.88; David B. Page v. R. William Hotchkiss, 400 N.E.2d 1255, Superior Court of Connecticut, Windham, 2.12.03.

¹⁷⁰ Andersen Consulting LLP. v. Gavin, 26 Conn. L. Rptr. 496, Conn. Super. C., 3.2.00.

¹⁷¹ Schroders, Incorporated et al., Plaintiffs, v. Hogan Systems, Inc, 522 N.Y.S.2d 404; Arlington Electrical Construction Appellant v. Schindler Elevator Corporation, 1992 Ohio App. Lexis 953; Stenograph Corp. v. Microcat Corp., 1990 WL 146754 N.D.Ill., 1990; Advent Systems Ltd. v. Unisys Corp. 925 F.2d (3rd Cir. 1991) 670 (673); Micro Data Base Systems v. Dharma Systems Inc. 148 F.3d (7th Cir. 1998) 649 (654)); Colonial Life Ins. Co. of America v. Electronic Data Services Inc., 817 F.Supp. 235 (D.N.H. 1993); andererseits Andersen Consulting LLP v. Gavin, 26 Conn. L. Rptr. 496 (Conn. Super.,

Stellungnahmen aus der Literatur¹⁷² stuft Softwareentwicklungsverträge gleichwohl als Verträge über „goods“ ein und wendet den U.C.C. an. Programme seien als Code, der den Computer zur Ausführung bestimmter Funktionen anweist, das Ergebnis eines intellektuellen Prozesses. Das Ergebnis dieses Prozesses werde dann zur Software, wenn es auf ein Medium übertragen wurde. Dieser Vorgang sei vergleichbar mit der Aufzeichnung von Musik auf einer CD oder dem Halten einer Vorlesung, die in Buchform übertragen wird. Sobald das Programm in Form einer Diskette oder einem anderen Medium vorliegt, ist es „tangible, moveable and available in the marketplace“¹⁷³.

Rechtspolitisch wird die Anwendbarkeit des U.C.C. zudem als wünschenswert angesehen, da damit ein einheitliches Recht bezogen auf einen weiten Umfang von Fragen, die üblicherweise in Computersoftware-Auseinandersetzungen auftauchen, zur Verfügung steht. Dies sei ein starkes Argument („strong policy argument“)¹⁷⁴ für den Rückgriff auf das Uniform Law¹⁷⁵.

In den von der Rechtsprechung entschiedenen Fällen werden neben der Lieferung von Software meist weitere Leistungen erbracht. Dann entscheidet der sog. predominant-factor-Test über die Einordnung¹⁷⁶: “The question becomes whether the dominant factor or essence of the transaction is the sale of the materials or the service¹⁷⁷.”

Im Ergebnis ist nicht sicher vorhersehbar, ob amerikanische Gerichte den U.C.C. anwenden oder nicht. Der Prozess der Meinungsbildung ist noch

2000), 2000 WL 225940; weitere Nachweise bei Scott, a.a.O., § 7.09[A], S. 7-30.

¹⁷² Nachweise unter *Advent Systems Ltd. v. Unisys Corp.*, 925 F.2d 670 (3rd Cir. 1991) S. 675f.

¹⁷³ *Advent Systems Ltd. v. Unisys Corp.*, a.a.O. S., 674f.

¹⁷⁴ *Advent Systems Ltd. v. Unisys Corp.* a.a.O. S. 676, ebenso *Systems Design and Management v. Kansas Credit Union*, 788 P.2d. 878 (Kan.App., 1990).

¹⁷⁵ Siehe auch oben S. 36, Fn. 130.

¹⁷⁶ *Richard A. Rosenblatt & Company, Inc., Appellant, v. Davidge Data Systems Corporation et al.*, 743 N.Y.S.2d 471; 2002 N.Y. App. Div. LEXIS 6154; *Propulsion Technologies, Inc., d/b/a PowerTech Marine Propellers, Plaintiff-Appellee-Cross-Appellant, v. Attwood Corp.*, 369 F.3d 896; 2004 US App. LEXIS 10429; *ePresence, Inc., Plaintiff v. Evolve Software, Inc.*, 190 F. Supp. 2d 159; 2002 US Dist. LEXIS 3742; *David B. Page v. R. William Hotchkiss*, 2003 Conn. Super. LEXIS 3341; 52 U.C.C. Rep. Serv. 2d (Callaghan) 365.

¹⁷⁷ *Incomm, Inc. v. Thermo-Spa, Inc.*, 41 Conn.Sup. 566, 570, 595 A.2d 954, 3 Conn. L. Rptr. 346 (1991).

nicht abgeschlossen¹⁷⁸. Je nach den Besonderheiten des Einzelfalles kann die Entscheidung zu Gunsten oder gegen seine Anwendbarkeit ausfallen¹⁷⁹.

4. Regeln zur Leistungsbestimmung

Ein "*contract for sale*" umfasst sowohl einen gegenwärtigen Verkauf als auch Verträge über zukünftige Verkäufe: „*both a present sale of goods and a contract to sell goods at a future time*“ (§ 2-106 (1) U.C.C.). Dabei besteht der Verkauf aus dem Übergang des Eigentums gegen Entrichtung des (Kauf-) Preises: „*A "sale" consists in the passing of title from the seller to the buyer for a price*“ (§ 2-106 (1) U.C.C.).

Die weitere Bestimmung der Leistung wird ausdrücklich den Vertragspartnern überlassen: „*Goods or conduct including any part of a performance are "conforming" or conform to the contract when they are in accordance with the obligations under the contract.*“ (§ 2-106 (2) U.C.C.). Aus deutscher Sicht interessant ist die Regelung hinsichtlich vorvertraglicher Vereinbarungen („*evidence of any prior agreement or of a contemporaneous oral agreement*“): Diese können grundsätzlich nicht für eine von dem Vertrag abweichende Bestimmung der Leistungspflicht herangezogen werden (sog. *Parol or Extrinsic Evidence-Rule*).

Anderes gilt für die Art und Weise der Vertragsdurchführung (*course of performance*¹⁸⁰), der bisher üblichen Gestaltung der Vertragsbeziehungen durch die Vertragsparteien (*course of dealing*¹⁸¹) sowie den für das jeweilige

¹⁷⁸ Auch in der Literatur wird das Thema anlässlich einer geplanten Revision des U.C.C. kontrovers diskutiert: Flechtner, a.a.O., S. 225 (226ff).

¹⁷⁹ Der endgültigen Entscheidung für oder gegen die Anwendbarkeit des U.C.C. wird in manchen Konstellationen (hier „downloadable products“) auch mit dem Argument ausgewichen, es gäbe hinsichtlich der Rechtsfolgen keinen essentiellen Unterschied zwischen U.C.C. Article 2 und dem Common Law of Contracts: SPECHT et al v. Netscape Communications Corp. & AOL Inc., 306 F.3d 17, US Ct. App. 2 Circt., 1.10.02.

¹⁸⁰ Definiert in § 1-303 (a) U.C.C.: A "*course of performance*" is a sequence of conduct between the parties to a particular transaction that exists if: (1) the agreement of the parties with respect to the transaction involves repeated occasions for performance by a party; and (2) the other party, with knowledge of the nature of the performance and opportunity for objection to it, accepts the performance or acquiesces in it without objection.

¹⁸¹ Definiert in § 1-303 (b) U.C.C.: A "*course of dealing*" is a sequence of conduct concerning previous transactions between the parties to a particular transaction that is fairly to be regarded as establishing a common basis of understanding for interpreting their expressions and other conduct.

Geschäft anerkannten Handelsbräuchen (*usage of trade*¹⁸²). Diese können für eine Erklärung und Ergänzung der Parteivereinbarung herangezogen werden.

Sind für die Erfüllung Spezifikationen der Leistung durch eine Partei erforderlich, so sind diese „*in good faith and within limits set by commercial reasonableness*“¹⁸³ zu erbringen. Werden Mitwirkungshandlungen der einen Partei nicht oder nicht rechtzeitig erbracht, so wird die andere Partei hinsichtlich der daraus folgenden Verzögerungen der eigenen Leistung von nachteiligen Rechtsfolgen befreit und kann entweder mit der eigenen Erfüllung fortfahren oder die unterbliebene Mitwirkung als Vertragsbruch behandeln¹⁸⁴.

Darüber hinaus dienen insbesondere die Regelungen hinsichtlich der Einstandspflichten des Verkäufers, sowohl infolge ausdrücklichen (*express warranties* – § 2-313) als auch infolge konkludenten Verhaltens (*implied warranties* §§ 2-314 bis 2-315) der Klärung der Gesichtspunkte, unter denen Leistungspflichten konkretisiert werden können. § 2-316 U.C.C. räumt dabei ein weitgehendes Recht zur Abbedingung von konkludenten Einstandspflichten ein, die sich zudem auch aus der Vertragsdurchführung, der Gestaltung der Vertragsbeziehungen sowie den Handelsbräuchen ergeben kann (§ 2-316 (c)).

Wenn und soweit „*warranties*“ nicht ausgeschlossen sind, zeigt sich hier ein bedeutender Unterschied zur Rechtslage bei Nichtanwendbarkeit des U.C.C.: In diesem Fall ist der Maßstab für die Erfüllung „*due care*“, was jedenfalls nicht gleichbedeutend ist mit einer Verantwortlichkeit für das Ergebnis dieses „*due care*“, während bei Anwendbarkeit des U.C.C. die Einstandspflicht aus den „*implied*“ oder „*express warranties*“ abgeleitet werden kann.

¹⁸² Definiert in § 1-303 (c) U.C.C.: A "*usage of trade*" is any practice or method of dealing having such regularity of observance in a place, vocation, or trade as to justify an expectation that it will be observed with respect to the transaction in question. The existence and scope of such a usage must be proved as facts. If it is established that such a usage is embodied in a trade code or similar record, the interpretation of the record is a question of law.

¹⁸³ § 2-311 (1) U.C.C.; Options and Cooperation Respecting Performance; Good faith ist in § 1-201 (20) definiert als „*honesty in fact and the observance of reasonable commercial standards of fair dealing*.“

¹⁸⁴ § 2-311 (3) U.C.C.; die Hauptfolge der Behandlung einer dem gemäßen Unterlassung als Vertragsbruch ist die Option, Schadensersatz (neben der Leistung) zu verlangen,

II. Convention on Contracts for the International Sale of Goods (CISG)

Die United Nation Convention on Contracts for the International Sale of Goods (CISG) wurde anlässlich der 97. Diplomatischen UN-Konferenz der Vereinten Nationen in Wien am 11. April 1980 verabschiedet und ist für die Vereinigten Staaten seit dem 1. Januar 1988 in Kraft¹⁸⁵. Sie ist Bestandteil des materiellen Rechts und findet ohne gesonderte Vereinbarung jedenfalls dann Anwendung, wenn die Niederlassungen (places of business - Art. 10 CISG) beider Vertragsparteien in verschiedenen Staaten liegen, die gleichermaßen die Konvention ratifiziert haben¹⁸⁶.

Die Art. 1 bis 4 CISG regeln die Anwendbarkeit der Konvention. Nach Art. 1 der CISG ist das Übereinkommen auf „Kaufverträge über Waren zwischen Parteien anzuwenden, die ihre Niederlassung in verschiedenen Staaten haben (...)“. Es werden nicht nur Kaufverträge über schon existierende Waren („goods“) erfasst, sondern auch „contracts for the supply of goods to be manufactured or produced“ (Art. 3 Abs.1 CISG), also Kaufverträge über noch herzustellende Waren.

Eine Anwendung wird dann ausgeschlossen, wenn der überwiegende Teil der Vertragspflicht des Lieferanten in der Ausführung von Arbeiten oder anderen Dienstleistungen („supply of labor or other services“) liegt (Art. 3 Abs.2 CISG).

Es ist zunächst zu fragen, ob Software als Ware („good“) i.S. der CISG angesehen wird. In der US-Rechtsprechung findet sich hierzu keine verwertbare Stellungnahme¹⁸⁷.

Die in der Literatur hierzu geführte Diskussion hat sich mit denselben Argumenten wie bei der Frage nach der Anwendbarkeit des U.C.C.

¹⁸⁵ Der CISG wird in der amerikanischen Literatur auch einfach unter der Bezeichnung „United Nations Convention“ und im deutschen Sprachraum als UN-Kaufrecht geführt, vgl. zur Terminologie: Corvaglia, Stefano; Das einheitliche UN-Kaufrecht, Bern, 1998.

¹⁸⁶ Art. 1 CISG.

¹⁸⁷ Die etwa gelegentlich in Bezug genommene Entscheidung *Evolution Online Sys. Inc. v. Koninklijke Nederland N.V.*, 2d Cir. 1998, 41 F Supp 2d 44, gibt keine Aufschlüsse über die hier interessierende Fragestellung der Anwendbarkeit der UN-Konvention auf Softwareentwicklungsverträge.

auseinander zu setzen. Solange Software in einem körperlichen Medium vorliegt, kann sie als Transformation von unkörperlichen Ideen und Wissen in eine physische Form verstanden werden¹⁸⁸. Der Benutzer erhalte etwas, und dieses „Etwas“ solle von der Konvention erfasst werden¹⁸⁹. Dem Hauptziel der Konvention, der „economic expansion and facilitation of international commerce“ sei damit zudem am besten gedient. Diese Auffassung wird unter dem Gesichtspunkt geteilt, dass sich der Erwerb von Software für viele, vielleicht die meisten Menschen, soweit sie sich überhaupt mit solch esoterischen Angelegenheiten befassen, als ein Kauf (*sale of goods*) darstellt. Es könne nicht erwartet werden, dass von der betroffenen Person verstanden wird, die Essenz des Vertrages liege im Recht zur Nutzung der Software¹⁹⁰. Dies Ergebnis ändere sich auch dann nicht, wenn die Software auf elektronischem Wege geliefert werde. In diesem Fall fehle es zwar an der Einbettung in einem körperlichen Medium, die Art der Übertragung sei jedoch auch hier für den Käufer unerheblich, und dem Ziel einer einheitlichen Rechtsanwendung komme auch nur eine breite Anwendung (der UN-Konvention) unter Einschluss der „electronic software“ entgegen¹⁹¹.

Allerdings könne dieser Standpunkt nicht auf Individualsoftware ausgedehnt werden. Software, die nach Vorgaben eines Bestellers speziell angefertigt werde, sei Gegenstand eines Dienstleistungsvertrages¹⁹². Eine endgültige Bewertung erfordere allerdings eine fallbezogene Analyse¹⁹³.

Im Ergebnis spielt die CISG für die rechtlichen Rahmenbedingungen von Softwareentwicklungsverträgen in den USA bisher allenfalls eine untergeordnete Rolle.

¹⁸⁸ Primak, Scott L.; Computer Software: Should the U.N. Convention on contracts for the international sale of goods apply ?; Computer Law Journal 1991, 197 (S. 215).

¹⁸⁹ „To the extent that the end-user is getting *something* that *thing* should be governed by the Convention.“ (Hervorhebungen im Original); Primak, a.a.O., S. 231.

¹⁹⁰ Carter, J. W.; Article 2B: International Perspectives; Journal of Contract Law 1999, 54 (S. 66).

¹⁹¹ Cox, Trevor; Chaos versus uniformity: the divergent views of software in the International Community; Business Law International 2000, S. 359.

¹⁹² Cox, a.a.O.; S. 359; Scott, a.a.O., S. 8-36.

¹⁹³ Scott, a.a.O., S. 8-41.

III. Uniform Computer Information Transaction Act (UCITA)

Der Uniform Computer Information Transaction Act, ursprünglich als Art. 2B U.C.C. konzipiert, ist trotz der jahrelangen Entwurfsbemühungen in der Praxis des amerikanischen Rechts von untergeordneter Bedeutung. Das Modellgesetz wurde bisher lediglich von 2 Staaten, Virginia und Maryland, in Kraft gesetzt. Über diese verschwindend geringe Akzeptanz hinaus zeichnet sich eine Gesetzgebung ab, die den UCITA-Bestimmungen zielgerichtet entgegenwirkt: Vermont, Iowa, North Carolina und West-Virginia haben so genannte „UCITA bomb shelter“ Gesetze verabschiedet, die Vereinbarungen von Gerichtsstand und Rechtswahl, sofern diese zur Anwendbarkeit des UCITA führen würden, für nicht durchsetzbar (*unenforceable*) erklären. Schließlich wurde das „Standby-Komitee der NCCUSL“¹⁹⁴ im Sommer 2003 aufgelöst, womit die Bemühungen darum, eine Einführung des Modellgesetzes durch weitere Bundesstaaten zu erreichen, praktisch aufgegeben sind¹⁹⁵.

D. Common Law

I. Restatements

Gem. Rs. 2d § 234 (2) gilt die Grundregel, dass, soweit nicht anders vereinbart, in Verträgen, die Erfüllungsleistungen von ausgedehnter Zeitdauer zum Gegenstand haben, zunächst diese zu erfüllen sind, bevor die andere, sofort erbringbare Leistung (Bezahlung) verlangt werden kann. Diese Regel erlangt vor allem in Dienstleistungsverträgen (*service contracts such as employment contracts*) Bedeutung. Sie findet allerdings nur Anwendung, wenn und soweit sich aus der Parteivereinbarung oder den Umständen nichts anderes ergibt. Ausreichend ist dabei, d.h. für die Fälligkeit der Gegenleistung, eine Erfüllung „im wesentlichen“ (*substantial performance*).¹⁹⁶

¹⁹⁴ National Conference of Commissioners on Uniform State Laws.

¹⁹⁵ Flechtner, a.a.O., S. 225 (S. 227).

¹⁹⁶ Calamari/Perillo; a.a.O., 11.17 und Fn. 3 (S. 412).

Der Maßstab für die Erfüllung ergibt sich zunächst aus Rs. 2d § 235 (1), wonach (nur) die volle Erfüllung zu einem Erlöschen der Leistungspflicht führt¹⁹⁷. Ist die Erfüllung fällig, so stellt jedes Abweichen einen Vertragsbruch dar¹⁹⁸, aus dem ein Recht auf Schadensersatz erwächst¹⁹⁹. Dies gilt selbst dann, wenn sich ein Schaden nicht beweisen lässt oder überhaupt kein Schaden entstanden ist, wobei dann allerdings allenfalls eine nominale Summe zugesprochen werden kann²⁰⁰.

Auf den Grad des Verschuldens kommt es nicht an, wenngleich die Gerichte unerhebliche Abweichungen (*trifling departures*) ignorieren können²⁰¹.

Geringfügige Mängel (*trivial defects*) sind daher in der Regel unbeachtlich²⁰².

Dies erweckt zunächst den Eindruck einer strengen Einstandspflicht für die in einem Vertrag vereinbarten Konditionen. Wenn beispielsweise bei der Errichtung eines Hauses für 50.000 \$ die Kücheneinrichtung versehentlich abweichend vom Vertrag ausgeführt wird, was sich für 100 \$ ändern ließe, ansonsten aber eine vertragsgemäße Ausführung vorliegt, so stellt dies gleichwohl einen Vertragsbruch dar²⁰³.

Selbst bei Kaufverträgen, in denen nach der *perfect tender rule* eine uneingeschränkte Übereinstimmung der Ware mit den vertraglichen Vereinbarungen gefordert wird, wendet die Rechtsprechung allerdings regelmäßig die Doktrin der sog. „*substantial performance*“ an²⁰⁴.

¹⁹⁷ „Full performance of a duty under a contract discharges the duty.“

¹⁹⁸ Rs. 2d § 235 (2): „When performance of a duty under a contract is due any non-performance is a breach.“

¹⁹⁹ Rs. 2d § 346 (1): „The injured party has a right to damages for any breach by a party against whom the contract is enforceable unless the claim for damages has been suspended or discharged.“

²⁰⁰ Rs. 2d § 346 (2): „If the breach caused no loss or if the amount of the loss is not proved under the rules stated in this Chapter, a small sum fixed without regard to the amount of loss will be awarded as nominal damages.“

²⁰¹ Rs. 2d § 235, Comment a.

²⁰² Calamari/Perillo fassen dies unter dem Grundsatz: *de minimis non curat lex*, a.a.O., S. 418.

²⁰³ So die Illustration 2. im Kommentar zu § 235 Rs. 2d.

²⁰⁴ Calamari/Perillo, a.a.O., S. 421, vgl. auch Art. 25 CISG (als Teil des amerikanischen law of statutes): „A breach of contract committed by one of the parties is fundamental if it results in such detriment to the other party as substantially to deprive him of what he is entitled to expect under the contract, unless the party in breach did not foresee and a reasonable person of the same kind in the same circumstances would not have foreseen such a result.“

Bei Verträgen, die sich nicht im Austausch von Ware und Bezahlung erschöpfen – wie beispielsweise Bauverträgen - wird generell ein milderer Maßstab angelegt. Hier ist „*substantial performance*“ grundsätzlich ausreichend. Dieses Prinzip spiegelt sich etwa in Rs. 2d § 237 wider. Danach ist (für das Schicksal der Gegenleistungspflicht) zunächst zu fragen, ob überhaupt ein „*failure of performance*“ vorliegt²⁰⁵.

Beantwortet wird dies auf der Grundlage der Parteivereinbarung, daneben aber auch auf der ergänzender Regeln wie beispielsweise derjenigen hinsichtlich ausgelassener Bestimmungen („*omitted essential terms*“) und dem Grundsatz von „*good faith and fair dealing*“²⁰⁶. Insbesondere die erste Regel ist aus deutscher Sicht interessant: Haben sich die Parteien im Hinblick auf eine grundlegende (*essential*) Bestimmung nicht geeinigt, sei es, weil sie eine Frage nicht bedacht oder sie auch bewusst als unerheblich oder unerfreulich angesehen haben, so kann das Gericht (!) eine den Umständen angemessene Regel hinzufügen²⁰⁷. Dies gilt laut Kommentar z.B. selbst dann, wenn der Vertrag keine Regelung über den Preis enthält²⁰⁸.

Good faith and fair dealing ist ein im US-amerikanischen Recht mittlerweile anerkanntes Konzept²⁰⁹. Seine Bedeutung und der jeweils angemessene Rechtsbehelf wird von den Umständen abhängig gemacht²¹⁰. Hinsichtlich der Leistungsanforderungen werden zwei Typen von *bad faith* ausdrücklich genannt: *Abuse of a power to specify terms* und *interference with or failure to cooperate in the other party's performance*²¹¹.

²⁰⁵ Rs. 2d. § 237 b).

²⁰⁶ Rs. 2d. § 237 b).

²⁰⁷ Rs. 2d § 204: „*When the parties to a bargain sufficiently defined to be a contract have not agreed with respect to a term which is essential to a determination of their rights and duties, a term which is reasonable in the circumstances is supplied by the court.*“

²⁰⁸ Rs. 2d § 204 d.; in diesem Fall ist ein „*reasonable price at the time for delivery*“ festzusetzen, wie es auch wörtlich in § 2-305 (1) U.C.C. heißt.

²⁰⁹ Vgl. Calamari/Perillo a.a.O., 11.38 S. 457ff, und § 1-304 U.C.C.: „*Every contract or duty within [the Uniform Commercial Code] imposes an obligation of good faith in its performance and enforcement.*“ Definiert wird *good faith* in § 1-201 (20) U.C.C. so: „*honesty in fact and the observance of reasonable commercial standards of fair dealing.*“ Etwas literarischer wird sie auch als die *Regel of the pure heart and the empty head* bezeichnet.

²¹⁰ Rs. 2d § 205 c.

²¹¹ Rs. 2d § 205 d.

Ergibt sich anhand der vorstehend genannten Kriterien, dass eine *failure of performance* vorliegt, so ist in einem zweiten Schritt zu fragen, ob diese sich als „*material*“ darstellt. Dafür finden sich Kriterien in Rs. 2d § 241²¹².

Ob Verzug diese Voraussetzungen erfüllt, ist eine der dabei am häufigsten diskutierten Fragen²¹³. Im Ergebnis erfüllt maßvoller (*reasonable*) Verzug diese Voraussetzung nur dann, wenn sich die Intention der Parteien zu dieser Rechtsfolge aus dem Vertragstext und den Begleitumständen ergibt²¹⁴.

Wird *material failure of performance* bejaht, so hängen die daraus abzuleitenden Rechtsfolgen davon ab, ob von der verletzten Partei noch erwartet werden kann, eine verspätete Leistung anzunehmen, oder ob sie von ihren eigenen Pflichten befreit wird²¹⁵. Auch eine schwerwiegende Vertragsverletzung berechtigt demnach nur dann zur Kündigung, wenn eine „Heilung“ nicht zumutbar ist.

Eine besondere Bedeutung im Rahmen dieser Untersuchung kann sich aus Rs. 2d § 208²¹⁶ ergeben. Danach hat das Gericht bei Klauseln, die es für den Zeitpunkt des Vertragsschlusses für unvertretbar (*unconscionable*)²¹⁷ hält, die Wahl, ob es den Vertrag oder Teile davon für wirksam erklärt oder die Anwendung des unvertretbaren Teiles so beschränkt, dass ein unvertretbares Ergebnis vermieden wird. Praktisch bedeutsam ist vor allem

²¹² Es werden u.a. genannt: Das Ausmaß, in dem die verletzte Partei des Vorteils verlustig geht, den sie vernünftigerweise erwarten konnte, eine angemessene Kompensationsmöglichkeit, die Wahrscheinlichkeit einer Abhilfe durch den Verletzer, sowie das Ausmaß des Verlusts, den der Verletzer erleidet. Konkretere Kriterien sind beispielsweise der Projektfortschritt - je früher der Vertragsbruch, um so eher ist er *material* - und der Verschuldensgrad, Calamari/Perillo, § 11.18 S. 414f.

²¹³ Calamari/Perillo, § 11.18 S. 415f.

²¹⁴ Der Mechanismus, durch die Formulierung „time is of the essence“ alleine diesen Effekt zu erreichen, funktioniert nach im Wachsen begriffener Ansicht nicht, Calamari/Perillo a.a.O., S. 416.

²¹⁵ Rs. 2d § 242.

²¹⁶ § 208. Unconscionable Contract or Term *If a contract or term thereof is unconscionable at the time the contract is made a court may refuse to enforce the contract, or may enforce the remainder of the contract without the unconscionable term, or may so limit the application of any unconscionable term as to avoid any unconscionable result.*

²¹⁷ Legt man die überkommen Formulierung vom „Anstandsgefühl aller billig und gerecht Denkenden“ (Mot. II 727) zugrunde, so entspräche der Begriff der Unconscionability der Sittenwidrigkeit etwa in § 138 BGB – Ein Vertrag erfüllt diese Voraussetzung, wenn „*such as no man in his senses and not under delusion would make on the one hand, and as no honest and fair man would accept on the other*“, Rs. 2d § 208 b.

die Möglichkeit, auf Grundlage der *Unconscionability* das Erzwingen der Vertragsdurchführung (*specific performance*) zu verweigern²¹⁸.

Schließlich findet sich in den Restatements die *Parol evidence rule*, wonach eine verbindliche Vereinbarung grundsätzlich alle vorhergehenden Vereinbarungen für unwirksam erklärt²¹⁹. Wie im U.C.C. werden auch hier *Usage of trade* und *Course of dealing* als Möglichkeit für die Ergänzung oder Qualifizierung der Parteivereinbarung genannt²²⁰.

II. Case Law

Naturgemäß finden sich im Fallrecht nur Reaktionen auf fehlgeschlagene Softwareprojekte. Welche Rolle die (detaillierte) Regelung der Leistungspflicht spielt, zeigt sich vor allem dann, wenn entsprechende Vereinbarungen nicht getroffen werden. So wurde es bereits 1984 in einem Prozess, der wegen mangelhafter Software von einem erfolgreich Schadensersatz begehrenden Kläger geführt wurde, als „usual industry practice“ bezeichnet, dass vor Beginn der Programmierung detaillierte Spezifikationen in Verbindung mit vertieften Besprechungen mit dem Kunden vorbereitet werden²²¹. Die Kommunikation mit dem Besteller in die richtigen Bahnen zu leiten, wurde - jedenfalls in dieser frühen Phase der Softwareentwicklung, als sich die entsprechende Industrie und der Aufbau von unternehmensinternen EDV-Abteilungen noch in der Entwicklung befand - als eine zentrale und in der Softwareentwicklung alltägliche Aufgabe des

²¹⁸ „Perhaps the simplest application of the policy against unconscionable agreements is the denial of specific performance where the contract as a whole was unconscionable when made.“ Rs. 2d § 208 g.

²¹⁹ Rs. 2d § 213. Als materielles Recht stellt es nicht etwa eine Auslegungsregel dar, sondern definiert den Gegenstand der Auslegung - Anmerkung a. Die Restatements of contracts geben dabei Detailregelungen an die Hand, was das Verhältnis der vertraglichen und außervertraglichen Umstände anbelangt (§§ 214-218).

²²⁰ Rs. 2d § 222 und § 223, vorbildlich ist diesbezüglich die Klarheit, in der § 203 die Prioritäten bei der Auslegung verdeutlicht: § 203 Standards of Preference in Interpretation:

In the interpretation of a promise or agreement or a term thereof, the following standards of preference are generally applicable:

(a) an interpretation which gives a reasonable, lawful, and effective meaning to all the terms is preferred to an interpretation which leaves a part unreasonable, unlawful, or of no effect;

(b) express terms are given greater weight than course of performance, course of dealing, and usage of trade, course of performance is given greater weight than course of dealing or usage of trade, and course of dealing is given greater weight than usage of trade;

(c) specific terms and exact terms are given greater weight than general language;

(d) separately negotiated or added terms are given greater weight than standardized terms or other terms not separately negotiated.

²²¹ Stahl Management Corp. v. Conceptions Unlimited; 554 F.Supp. 890 (1983).

Softwarehauses angesehen²²². Wurde ursprünglich noch davon ausgegangen, dass selbst größere Unternehmen wegen eines Wissensgefälles zu Gunsten des Softwarehauses schützenswert sein können²²³, so wurden mit zunehmender Verbreitung der EDV und damit der Softwareentwicklung auch an den Besteller höhere Anforderungen gestellt²²⁴. Auch ohne ausdrückliche dahingehende Vereinbarung kann sich ein Besteller nicht auf unzureichende Resultate des Softwarehauses berufen, wenn er seine Mitwirkungspflichten verletzt²²⁵. Die Herleitung dieses Ergebnisses nach dem Recht von Illinois beruhte auf einem Sechs-Stufen-Test für *breach of contract*²²⁶. Auf der vierten Stufe wird vom Kläger verlangt, dass er beweist, alle eigenen vertraglichen Verpflichtungen eingehalten zu haben. Auch Nachforderungen, d.h. das Bestehen auf Leistungen, die vom ursprünglichen Auftrag nicht gedeckt sind, können im Konfliktfall als Vertragsverletzung ausgelegt werden²²⁷. Beeinträchtigen die Änderungswünsche des Bestellers den Projektfortschritt, so kann er sich nicht unter Bezugnahme auf den Vertragsbruch des Softwarehauses von

²²² (...) when you are talking to non-computer people they talk in generalities and great vagueness because they just don't understand the details that are involved in automating anything. The job of the systems analyst is to listen to them, in their terms, and then to make an effort to convert that into the degree of precision that is required to write specifications that are adequate to write a computer program which is precise.“

Georgetown School of Arts and Sciences v. Microsystems Engineering Corp.; 984 WL 564182 (1984). Das Verfahren wurde allerdings zu Gunsten des Softwarehauses entschieden, weil der Besteller keine wesentlichen Fehler nachweisen konnte.

²²³ Dreier Company Inc. v. Unitronix Corp. And Andrew Yassenchak, 527 A. 2D 876 (1986) mit Verweis auf Hundred East Credit Corp. v. Eric Schuster, 212 N.J. Super. 357. Hier wurde das klagende Unternehmen in den Anwendungsbereich des Konsumentenschutzes einbezogen, da : „Even the most world-wise business entity can be inexperienced and uninformed in a given consumer transaction. Unlawful practices thus can victimize business entities as well as individual consumers.“

²²⁴ So wurde das Vorbringen des Klägers, er sei nicht erfahren genug gewesen, in einem Verfahren bzgl. der Anwendbarkeit einer vertraglichen Haftungsausschlussregelung des Softwarelieferanten als unglaublich zurückgewiesen: „Buyer was hardly the sheep keeping company with wolves that it would have us believe.“ Colonial Life Ins. Co. of America v. Electronic Data, 817 F. Supp. 235 (1993).

²²⁵ So wurde die Klage eines Bestellers vollständig abgewiesen, weil er kein ausreichendes Personal im Projekt beschäftigte, einen „key“-Mitarbeiter entließ und die Dateneingabe vernachlässigte: Baeco Plastics Inc. v. Inacomp Financial Services 1993 WL 410066 (1993).

²²⁶ Baeco Plastics Inc. v. Inacomp, a.a.O., „(...) a plaintiff must prove the following elements:(1) an offer and acceptance; (2) consideration, (3) the terms of the contract; (4) plaintiff's performance of all contractual conditions required of him; (5) defendant's breach of the terms of the contract; and (6) damages occurring as a result of the defendant's breach.“

²²⁷ Der Besteller hatte gekündigt und wurde wegen unstreitiger Nachforderungen und nicht erbrachter Mitwirkungsleistungen zunächst per Juryspruch zu Schadensersatz in Höhe von 2,6 Mio. \$ verurteilt, die das Gericht auf 469.000 \$ herabsetzte: Computrol Inc. v. Newtrend, L.P. And CA Newtrend, 203 F. 3D 1064 (2000).

seinen Verpflichtungen lösen²²⁸. Erbringt das Softwarehaus andererseits ohne konkreten Auftrag über die ursprüngliche Vereinbarung hinaus Leistungen, so ist es für die Vereinbarung einer dementsprechenden Vergütung selbst dann beweispflichtig, wenn der Besteller nicht im erforderlichen Maße mitarbeitet²²⁹. Die Tatsache alleine, dass eine Festpreisvereinbarung vorliegt, schließt die Vergütungspflicht für Zusatzleistungen jedenfalls dann nicht aus, wenn die von dem Softwarehaus vorgenommenen Arbeiten nicht in den Aufgabenbereich des Vertrages fallen²³⁰. In der jüngeren Rechtsprechung spiegelt sich die Reaktion der Softwarehäuser wider, den Besteller ausdrücklich mit in die Pflicht für den Projekterfolg zu nehmen und die Verantwortlichkeit für Misserfolge im Projekt (auch) auf den Besteller zu übertragen. So finden sich beispielsweise in den Erstellungsverträgen Bestimmungen, wonach der Besteller auf Anforderung des Softwarehauses zeitgerecht Material vorzulegen hat. Dies ist zum einen als Bedingung für die Leistungspflicht des Softwarehauses gestaltet und führt zum anderen bei Verstoß zu einer Ausdehnung des dem Softwarehaus eingeräumten Zeitrahmens²³¹. Weiterhin wird der Versuch unternommen, den Rechtsbehelf bei nichtvertragsgemäßer Beschaffenheit auf Reparatur,

²²⁸ Einem Softwarehaus wurden ca. 42 Mio. \$ Schadensersatz zugesprochen, weil der Staat California Änderungen im Umfang von ca. „1,500,000 lines of Software code“ in Auftrag gab, den Vertrag dann kündigte und für die Änderungsarbeiten nicht zahlen wollte: *State of California v. Lockheed Martin IMS*, 2002 WL 99554 (2002).

²²⁹ Der Besteller wollte auf Basis einer mündlichen Absprache die Bezahlung von Zusatzleistungen erzwingen. Das Gericht wies dies zurück: *Objectwave must show that there has been "an offer, an acceptance, consideration, and sufficient specification of terms so that the obligations involved can be ascertained."* *Savoca Masonry Co., Inc. v. Homes and Son Constr.*, 112 Ariz. 392, 394, 542 P.2d 817, 819 (1975). While contract terms need not be set forth in minute detail, necessary terms frequently include time of performance, place of performance, price or compensation, and penalty provisions. *Pyeatte v. Pyeatte*, 135 Ariz. 346, 350, 661 P.2d 196, 200 (Ariz.Ct.App.1982). *Objectwave Corp. v. Authentix Network Inc.* 2002 WL 1477625.

²³⁰ *Northrop Grumman Corp. v. The United States*; 42 Cont. Cas. Fed. (CCH) P77,381 (1998).

²³¹ *IXL inc. v. AdOutlet.com Inc.*, 2001 US Dist. LEXIS 3784 (2001) – AdOutlet war hier der Besteller, iXL das Softwarehaus: "*AdOutlet shall perform the tasks set forth in this Statement, as a condition to iXL's obligations to perform hereunder,*" and that at iXL's request AdOutlet shall provide client materials "*required by iXL to perform the tasks set forth in the Statement*" (Id., P 2.2). AdOutlet "*shall comply in a timely manner, with all reasonable requests of iXL for assistance in enabling iXL to fulfill any of iXL's obligations under this Statement.*" A failure by AdOutlet to meet its deadlines will result in an extension of the time allowed for iXL to perform. Der Vertrag gab dem Softwarehaus das Recht, bei Verstoß des Bestellers gegen seine (Mitwirkungs-) Pflichten zu kündigen, und dies bei fortbestehender Zahlungspflicht für alle bis zur Wirksamkeit der Kündigung ausstehenden Arbeiten: "*The Agreement further provided for termination upon other material defaults of duties and obligations under the Agreement (other than payment obligations), if there was a failure to substantially cure or commence a cure of the default within thirty days of receiving written notice. (...) AdOutlet "shall be obligated to pay iXL for all services rendered pursuant to any outstanding Statements of Work through the effective date of such termination."*

Korrektur oder Austausch zu beschränken²³². Schließlich kann die Pflicht zur Spezifikation der Leistung dem Besteller auferlegt werden. Dann führt eine mangelhafte Spezifikation zur Entlastung des Softwarehauses²³³.

E. Zusammenfassung und Ergebnis

Die Leistungspflicht ist im amerikanischen Recht nur teilweise vertragsspezifisch, d.h. für einen bestimmten Vertragstyp angepasst geregelt. An einer dem deutschen besonderen Schuldrecht vergleichbaren gesetzlichen Regelung für die Erstellung eines Werkes fehlt es. Als Rechtsquellen kommen für den Untersuchungsgegenstand Gesetzesrecht (*law of statutes*) und Fallrecht (*Common Law*) in Betracht. Im Bereich des *law of statutes* sind der U.C.C., die CISG und der UCITA zu nennen. Im Bereich des *case law* sind neben den eigentlichen Entscheidungen die Restatements of Law als vielbeachtete Autorität heranzuziehen. Die häufig anzutreffenden Parallelen von Restatement on the Law of Contracts und dem U.C.C. begünstigen ein einheitliche Rechtsprechung. Die Anwendung der Restatements wird häufig die indirekte Anwendung des U.C.C. zur Folge haben²³⁴. Dieser Effekt und die Tatsache, dass hier über viele Jahre hinweg eine Vielzahl von führenden amerikanischen Juristen maßgeblich mitgewirkt hat²³⁵, unterstreichen die Bedeutung der Restatements als Referenz von kaum überschätzbarer Tragweite.

Der U.C.C. dient in erster Linie als Auffangregelung. Ob er auf Softwareentwicklungsverträge angewendet werden kann, ist - in erster Linie

²³² "Risetime's "sole and exclusive remedy for any noncompliance with the warranty will be limited to repair, correction, or replacement of the nonconforming aspect of the licensed software by CCI." Risetime Inc. v. Colorado Customware Inc. 2003 US Dist. LEXIS 19470 (2003). Das Gericht überantwortete die Entscheidung einer Jury, da es sich um eine tatsächliche Frage (der Vertragsauslegung) handele.

²³³ Die Klage des Bestellers auf Rückzahlung der an das Softwarehaus entrichteten Vergütung wurde abgewiesen, da die Mangelhaftigkeit der Software nicht nachgewiesen sei, bzw. die Fehler nicht nachweislich auf Mangelhaftigkeit der Software beruhten: "Some of the problems resulted from Sutter's failure to acquire the proper type of printer; some resulted from what turned out to be the unanticipated inadequacy of the specifications that Sutter had prepared and accepted; and some were attributable, quite simply, to the ordinary process of adjustment to a new system." Sutter Insurance Co. v. Applied systems Inc. 2004 US Dist. LEXIS 3802 (2004).

²³⁴ Corbin, a.a.O., S. 86.

²³⁵ Als drei für das US Contract Law richtungsweisende Juristen sind hier zu nennen Professor Williston, Professor Arthur L. Corbin and Professor E. Alan Farnsworth, Calamari/Perillo, a.a.O., 1.6 S. 15f.

wegen der Schwierigkeit bei der Einordnung von Software in das Rechtssystem - sehr umstritten. Nicht selten führen rechtspolitische Erwägungen zu seiner Anwendung. Dies ermöglicht es, sich auf die vertrauten Rechtsinstitute der *implied* und *express warranties* zu beziehen, wobei der U.C.C. insgesamt in hohem Maße die „Parteihohheit“ bei der Leistungsbestimmung anerkennt, wie sich z.B. bei der Möglichkeit der Haftungsfreizeichnung (*Disclaimer*) zeigt.

Die Definition der Leistung ist nichts anderes als die Beschreibung dessen, „was ich will“. Das Recht der USA geht von dem Grundsatz der Vertragsfreiheit aus. Die Vertragspartner können im Rahmen des Gesetzes nach eigenem Gutdünken vertragliche Bindungen eingehen²³⁶. Beschränkungen ergeben sich aus der staatlichen Sozialkontrolle, vor allem zu Gunsten eines (potentiell) schwächeren Vertragspartners (Minderjährigen, Angestellten, Mieters, Verbrauchers etc.)²³⁷. Softwareentwicklungsverträge werden in der ganz überwiegenden Zahl zwischen Parteien geschlossen, die keiner der typischerweise geschützten Gruppen angehören. Die Grenzen sind daher weit gesteckt.

Eine wichtige Rolle bei der Ermittlung der geschuldeten Leistungen können die Vertragsdurchführung und -gestaltung sowie die Handelsbräuche spielen. Dies gilt vor allem für die Auslegung sowie das Einbeziehen oder den Ausschluss von Einstandspflichten.

Die CISG und der UCITA leisten keinen merklichen und verwertbaren Beitrag für die hier relevanten Fragestellungen.

Auch die Restatements rücken die Parteivereinbarung und Hilfsmittel dafür, wie diese ggf. ermittelt werden kann, in den Vordergrund. Dabei besteht ein weitgehendes Ermessen des Gerichtes für die Beurteilung von Verträgen. Es erwächst aus einer Ergänzungskompetenz für Vertragslücken und dem Ausfüllen der unbestimmten Rechtsbegriffe von *good faith and fair dealing*, sowie der *unconscionability*. Die Ausübung des Ermessens muss sich

²³⁶ Sie haben m.a.W. das Recht, „to call forth legal reinforcement of exchange relations“, Macneil/Gudel, a.a.O., S. 403; siehe auch Calamari/Perillo, a.a.O., S. 4ff.

²³⁷ Vgl. Macneil/Gudel, a.a.O., S. 402ff.

allerdings an der Vertragsdurchführung und den Handelsbräuchen orientieren. Die Höhe der Messlatte für ordnungsgemäße Erfüllung der Leistungsanforderungen wird in aller Regel durch *substantial performance* bestimmt.

In der Rechtsprechung zeichnet sich die Tendenz ab, dem Besteller mehr Verantwortung für das Erreichen des Vertragsziels zu übertragen. Je deutlicher es wird, dass Softwareentwicklungsverträge keine einfachen Austauschverträge von Ware und Geld darstellen, um so differenzierter fällt die Beurteilung der Verursachungsbeiträge im Falle eines Scheiterns aus. Die Leistungsbestimmung wird den Risikosphären beider Parteien zugeordnet, und die Gerichte prüfen im Zweifelsfall anhand der konkreten Vereinbarungen und der Projektgeschichte genau nach, wie die Verantwortungsbereiche abgegrenzt wurden und welche Beiträge die Parteien erbracht bzw. nicht erbracht haben.

Kapitel 4: Empirischer Befund in den USA

Die ersten Kapitel haben gezeigt, dass es aus technischer und rechtlicher Sicht kein einfaches Unterfangen darstellt, die Leistung in einem Softwareentwicklungsvertrag zu bestimmen. Da die Informationstechnologie und mithin die Produktion von Software ein bedeutender Industriezweig ist, werden vielfach entsprechende Verträge geschlossen. In diesem Kapitel werden sieben Musterentwürfe und drei Beispiele aus der Praxis untersucht.

A. Musterentwürfe

Die Untersuchung basiert auf sieben Musterentwürfen von Raysman/Brown (RB), David F. Simon (DaS), Amy Slater (AS) und Diane W. Savage (DS) sowie einem in der Datenbank nicht genannten Autor (FL für Florida)²³⁸. RB und DaS unterscheiden zwischen Verträgen, die an den Interessen des Softwarehauses („*Vendor Oriented*“) und denen des Softwarebestellers („*User Oriented*“) orientiert sind²³⁹. Es wurden jeweils beide Varianten in die Untersuchung aufgenommen (im Folgenden als RB-V und RB-U bzw. DaS-V und DaS-U bezeichnet). Die Entwürfe von AS, DS und FL werden im Folgenden demgemäß auch zusammengefasst als „neutrale Entwürfe“ bezeichnet. Das Softwarehaus wird unterschiedlich bezeichnet: Als *Licensor* (RB-V und RB-U, der Softwarebesteller ist der *Licensee*), als *Vendor* (DaS-V und DaS-U, wobei der Softwarebesteller bei DaS-V *Customer* und bei DaS-U *User* heißt), oder als *Developer* (bei DS, AS und FL, wobei der Softwarebesteller bei DS mit XXXXX, bei AS als *client* und bei FL als *customer* bezeichnet wird).

²³⁸ Raysman, Richard/Brown, Peter; *Computer Law: Drafting and Negotiating Forms and Agreements*, Lsbl. New York 2003; Simon, David F., *Computer Law Handbook*, Philadelphia, 1990; Slater, Amy, in: *California Transaction Forms*, Westlaw Database § 9:59, 2003; Savage, Diane W., in: *Nichols Cyclopedic of Legal Forms Annotated*, Westlaw Database, § 2.3191, 2002; Unbekannter Autor, in: *Florida Jur Forms Legal and Business Database* § 20B:1, 2003.

²³⁹ Bei Raysman/Brown ist das User-orientierte Muster auf eine grundsätzliche Neuerstellung, das Vendor-orientierte Muster dagegen auf die Anpassung einer bereits vom Softwarehaus entwickelten Software ausgerichtet.

I. Vertragsart

Aus der Vertragsbezeichnung lassen sich nur begrenzte Schlüsse für die Leistungsbestimmung ziehen. RB sprechen von *Software Development and License Agreement*, AS und DaS-U von *Software Development Agreement* bzw. *Custom Software Development Agreement*, DaS-V von *Software Development and License Agreement*, DS von *Software Development and Acquisition Agreement* und FL von einem *Agreement for Development of Software to Specifications*. In allen Mustern ist die Einräumung einer Lizenz vorgesehen, wobei AS, DaS-V, RB-U und RB-V keine Exklusivität vorsehen, während DS, DaS-U und FL exklusive, d.h. eine auch das Softwarehaus von der weiteren Nutzung ausschließende Rechteübertragung beinhalten. In den zwei letztgenannten Verträgen ist diese Exklusivität allerdings zeitlich auf ein Jahr (FL) bzw. 25 Jahre (DaS-U) begrenzt.

Im Muster von DS findet sich in der Präambel die Formulierung:

„2. XXXXX desires to engage Developer to perform certain software development services and Developer is willing to perform software development services for XXXXX upon the terms and conditions set forth herein.“

Vertragsinhalt ist demnach das Erbringen von „*Software development Services*“. Sehr ähnlich formuliert AS in der Präambel, indem er die *performance* von „*services*“ in den Vordergrund rückt:

„C. Client wishes to have Developer design and develop certain computer software according to the terms, specifications, and conditions set forth in this Agreement and Developer desires to perform these services.“

Unter der Überschrift *Development Obligations* trifft man dementsprechend auf die Formulierung:

2.1. Generally. Developer will provide the development Services and complete the work described in the scope of work.

Bei FL heißt es:

1.01. Development Undertaking. Developer shall, within _____ days of final execution of this Agreement, commence work on and diligently proceed with the development of the Subject Programs according to and in conformity with the specifications and performance standards set forth in Exhibit A (Specifications and Performance Standards).

Vom Softwarehaus ist hier nicht ein feststehendes Ergebnis, sondern die Arbeit im Hinblick auf dieses Ergebnis geschuldet.

Während sich also deutlich die *Service-Orientierung* als Vertragsinhalt bereits aus dem Vertragswortlaut ablesen lässt, ist die Vertragseinordnung in den übrigen Mustern zumindest hinsichtlich des Wortlautes nicht eindeutig.

Im Gegensatz zu RB-V, wo unter Nr. 2 davon die Rede ist, „... *Licensor hereby agrees to develop the Management System for Licensee...*“ formuliert RB-U

1. The Software System

1. 1 Definition of the System

Licensor agrees to supply to Licensee a data processing software system (the „System“) which meets Licensee’s requirements contained in Licensee’s „Request for Proposal“ as all such requirements may be modified or superseded by the Detailed Design Specifications as defined herein.

Die Betonung liegt auf der Lieferung eines *Systems*, welches den Anforderungen des Softwarebestellers entspricht.

Ähnlich unterschiedlich ist der Wortlaut bei DaS. DaS-V gibt dem Softwarehaus vor, in den Spezifikationen beschriebene (*nicht*: diesen Spezifikationen entsprechende) Programme zu entwickeln:

1. Software Development

Pursuant to the terms of this agreement, Vendor shall develop for Customer certain computer programs described in the Specifications set forth in Attachment „A“ and related user-level documentation (the „Software“), install the Software on Customer’s computer equipment, and make it ready for use. Customer acknowledges that it has carefully reviewed Attachment „A“, which fully sets forth all of customers requirements, and that Vendor’s compliance with it shall constitute satisfactory performance under this agreement.

DaS-U macht bereits in der Präambel deutlich, dass es um die Lieferung von an den Bedürfnissen und Anforderungen des Bestellers orientierter Software gehen soll:

„Vendor desires to provide application software to User that meets User’s needs and requirements.“

Muster, die sich am Interesse des Softwarehauses orientieren, bevorzugen also eine „Service“-Lösung, wohingegen die am Softwarebesteller ausgerichteten Muster die Lieferung (*supply, provide*) eines Systems präferieren.

II. Bestandteile der Lieferung

Was Bestandteil der Lieferung sein soll, ist bei AS explizit in dem Anhang „SCOPE OF WORK“ dargestellt:

- II. DELIVERABLES. Developer will deliver the following to Client:*
- 1. A description of the Software, including an overview of how the Software works; a flow diagram indicating the input, flow, processing, and output of information; and specifications for the Software, including minimum hardware and operating system requirements.*
 - 2. Complete user documentation, including a description of how to access and use the Software, screen prints of menus and input/output screens, data input descriptions, sample output/report forms, error code descriptions and solutions where appropriate, and explanations of all necessary disks and data used by the Software.*
 - 3. Complete program/technical documentation, including program source code listings with comments, technical information about files and their locations, file names, file/database structures, record structures, and layout and data elements.*
 - 4. A Master Copy of the Software on magnetic media, including all programs, on-line documentation, and any documentation developed on a computer.*

Gefordert ist also eine Sammlung technischer und erklärender Unterlagen, die den Entwicklungsvorgang von der Definition der Anforderungen bis hin zu der Kommentierung des Source-Code erfasst.

Bei DS findet sich hierzu unter 2. *Delivery and Acceptance Testing*:

„(a) Developer shall deliver the Specifications, Product and Documentation to XXXXX in accordance with the schedule set forth in Exhibit B. The term "Documentation" means all technical documentation for the Product,

including flowcharts, source code, object code, program procedures and descriptions (including descriptions of the source code and build procedures for the object code), procedures for maintenance and modification, testing data and similar written material relating to the design, structure and implementation of the Product. The term "Documentation" does not include end user documentation or manuals. The Documentation shall be sufficient to enable a reasonably skilled and experienced programmer to understand the design, structure and implementation of the Product."

Der letzte Satz trifft hier den Kern der vorstehenden Liste von Lieferpflichten:

Der Softwarebesteller soll durch sie in die Lage versetzt werden, die Software von diesem Zeitpunkt an - falls gewünscht oder notwendig - selbständig zu pflegen oder ändern. Er soll m.a.W. aus der Abhängigkeit von der Expertise des Softwarehauses in die Freiheit zur selbstbestimmten Nutzung und Entwicklung der Software gelangen.

DaS-U verwendet einen inhaltsgleichen Satz in seiner Beschreibung der zu liefernden *Documentation* unter 6. :

„Vendor shall deliver to User all necessary and reasonable documentation for the System, including user, systems, operating, and program manuals. Such documentation shall include, but not be limited to, ... [Anm.: Es folgt eine längere Aufzählung der technischen Dokumente].“

„The documentation (together with the source code) shall provide a competent programmer with sufficient information to maintain and enhance the system.“

In der am Softwarehaus orientierten Variante DaS-V findet sich die Tendenz, den Lieferumfang zu begrenzen bzw. zumindest klar zu definieren, indem eine vollständige Beschreibung zusätzlicher Software oder Dokumentation in Anhängen gefordert wird :

1. Software Development

(...) Any different or additional software or documentation developed by Vendor for Customer or services to be provided by Vendor to Customer shall be fully described in additional Attachments, which shall also be subject to the terms and conditions of this Agreement and such other terms and conditions as Vendor and Customer shall agree upon in each respective Attachment.

Der Umfang der Dokumentationslieferungspflicht ist durch die Begrenzung auf „user-level“ auf einem eher zurückhaltenden Niveau angesiedelt:

4. Training and Documentation

Vendor shall provide to Customer user-level documentation for the Software. Additional training and support services and documentation shall be provided to Customer upon Customer's written request at Vendor's then-current hourly rates plus expenses.

Bei FL ist die Lieferpflicht grundsätzlich an den Terminplan geknüpft:

2.01. Delivery and Installation. Immediately on the completion of each development milestone set forth at Exhibit B, Developer shall deliver and install the Subject Programs at the _____ [data-processing operations center of Customer] and shall deliver to Customer at that address all documentation and other materials required to be provided in accordance with the milestone.

RB-V und RB-U unterscheiden sich dahingehend, dass RB-V im wesentlichen eine Übertragung der Lizenz zum Gegenstand hat:

2. Development and License

Subject to all of the terms and conditions hereof, Licensor hereby agrees to develop the Management System for Licensee, and does hereby agree to grant to Licensee a perpetual, non-exclusive and non transferable license (the „License“) to use the Management System, terminable as specified herein, which license the Licensee hereby accepts.

während RB-U - neben der Lizenzeinräumung - die Lieferung eines „Systems“ verlangt, welches den Anforderungen des Softwarebestellers entspricht²⁴⁰. Es versteht sich fast von selbst, dass hierzu auch eine ausführliche Dokumentation gehört:

4.10. Documentation

The documentation to be provided on or before Preliminary Acceptance of each Application of the Software System shall include the Detailed Design Specifications, system flow charts, program flow charts, file layouts, report layouts, screen layouts, program source code, user instruction manuals, and all other documentation as specified in the RFP (Anm.: RFP steht für Request for Proposal) and the Licensor Proposal.

Unter dem Gesichtspunkt der Lieferung enthalten alle Muster die Verpflichtung des Softwarehauses, eine funktionsfähige Software bereitzustellen. In vier der sieben Muster sind nach dem Vorstehenden neben der Software selbst ausführliche Dokumentationsunterlagen an den Softwarebesteller auszuhändigen. In einem Muster wird der diesbezügliche

²⁴⁰ S.o. S. 61.

Lieferumfang der genaueren Bestimmung durch die Vertragspartner überlassen. In den Softwarehaus-orientierten Mustern muss neben der Software selbst nur in geringem Umfang (DaS-V) bzw. kein weiteres Material (RB-V) zur Verfügung gestellt werden.

III. Spezifikationen

Bis auf RB-V, das die Entwicklung auf der Basis von bereits existierender Software auf Seiten des Softwarehauses vorsieht, gehen alle Muster von einem mehrstufigen Verfahren zur Bestimmung des Vertragsinhaltes aus. Dies besteht theoretisch aus drei Schritten. Zunächst werden die Anforderungen an die Software formuliert. Diese Beschreibung wird in ein möglichst präzises Anforderungsprofil transformiert und schließlich im letzten Schritt programmtechnisch umgesetzt. Wer im Rahmen dieser Vorgehensweise welche Pflichten übernimmt, ist unterschiedlich geregelt. Bis auf RB-V gehen alle Muster davon aus, dass das Softwarehaus für den zweiten Schritt, die „Transformation des Wünschenswerten in Machbares“ verantwortlich ist. Die Details werden im folgenden dargestellt.

Bei AS heißt es unter

*2.2. **Technical Design Phase.** Developer will develop the Technical Design for the Software in accordance with the Scope of Work. Developer shall deliver the completed Technical Design on or before _____ [date].*

Das Softwarehaus soll demnach auf Basis der „*Scope of work*“ das technische Design der Software übernehmen. „*Scope of work*“ ist unter 1. definiert als:

1.6. "Scope of Work" means the document attached as Exhibit A to this Agreement that describes the Development Services, features, functions, and specifications of the Software, contents of the Documentation, and other information necessary to define the work to be performed pursuant to this Agreement.

Es bleibt offen, wer den Inhalt des „*Scope of work*“ erstellt.

DS unterscheidet unter 1. ausdrücklich zwei Phasen, nämlich die Erstellung der schriftlichen Produktspezifikation und die Entwicklung des Produktes:

Software Development Services.

(a) Developer agrees to design and develop the software product(s) described in the attached Exhibit A (the "Product(s)") in two phases consisting of the creation of written Product specifications (the "Specifications") and the development of the Product, pursuant to the schedule set forth in the attached Exhibit B.

Die genannte Anlage A ist dann - ohne weiteren Inhalt - mit „*Functional Requirements*“ betitelt. Wer diese funktionellen Anforderungen formuliert, ist im Muster nicht festgelegt.

Auch bei FL wird das Softwarehaus verpflichtet, die Software „*in conformity with the specifications and performance standards set forth in Exhibit A*“ zu entwickeln, ohne die Festlegung, wer die Spezifikationen erstellt.

Bei DaS zeigt sich eine deutliche Differenzierung hinsichtlich Softwarebesteller- und Softwarehaus-orientierter Variante. DaS-V spricht zwar nicht aus, wer die Anforderungen formuliert, auferlegt dem Softwarebesteller aber unter Nr. 1 S. 2 die Pflicht, eine entsprechende Anlage auf Vollständigkeit und Richtigkeit zu überprüfen:

Customer acknowledges that it has carefully reviewed Attachment „A“, which fully sets forth all of Customer’s requirements and that Vendor’s compliance with it shall constitute satisfactory performance under this agreement.

Anders stellt sich dies bei DaS-U dar. Hier wird das Softwarehaus im Rahmen der ersten Projektphase verpflichtet, die Bedürfnisse der Softwarebestellers zu eruieren:

2. Project Phases

(a) Phase I – System Definition

(i) The first task to be undertaken by Vendor is the investigation of all of User’s computer needs, and the preparation of a detailed written report summarizing those needs and the specifications for the system meeting those needs. Such specifications, to be identified by phase, hereinafter referred to as the „Design Document“, shall include but not be limited to, descriptions of functionality, response time for critical functions, input formats, record layouts, output reports, user manuals, operations manuals, programmer documentation (including narratives on programming conventions and structure), and screen layouts.

Bei RB-U findet sich dieser Gedanke in abgeschwächter Form wieder. Das Softwarehaus bleibt für die Ermittlung der notwendigen Daten verantwortlich, aber der Softwarebesteller wird zur Mitarbeit, insbesondere in Form der Bereitstellung von Personalressourcen berufen:

4.7. Detailed Design Specifications

A. For each Application Licensor shall, with Licensee's full cooperation, gather the necessary data and develop Detailed Design Specifications. Licensee shall make available personal knowledgeable in Licensee operations with respect to each Application for the number of days set forth on Schedule „H“ attached hereto.

Als Ausgangsdokument setzt RB-U dabei den sog. „Request for Proposal“ voraus, der vom Softwarebesteller als Grundlage für dessen Ausschreibung formuliert wurde²⁴¹.

Liegen die vom Softwarehaus erstellten Spezifikationen vor, folgt in einem Teil der Muster ein zweiter Schritt, in dem diese Spezifikationen vom Softwarebesteller daraufhin geprüft werden, ob sie seinen in den „functional requirements“ niedergelegten Vorstellungen entsprechen. Dies wird mit Fristen und weiteren Verfahrensregeln verknüpft:

Bei DS heißt es unter 1 (b):

(b) XXXXX will have the period set forth in Exhibit B to review the Specifications. XXXXX will report any deficiencies in the Specifications to Developer and Developer will correct the deficiencies within the period set forth in Exhibit B. XXXXX's acceptance or non-acceptance of the Specifications will be based on XXXXX's determination of whether the Specifications represent an acceptable implementation of the functional requirements set forth in Exhibit A (the "Functional Requirements"). If XXXXX accepts the Specifications, Developer will develop the Product in accordance with the Specifications and Functional Requirements.

Ähnlich findet sich bei AS unter 2.3:

2.3. Acceptance of Technical Design. Client has ten (10) days from the date on which Developer delivers the Technical Design in which to accept or reject it in writing. If Client rejects the Technical Design, Client must specify in writing Client's grounds for rejection and Developer will

²⁴¹ S.o. S. 61.

use its best efforts to revise the Technical Design to make it acceptable to Client within the following ten (10) days. If Client rejects the Technical Design a second time, Client will have the option of repeating the procedure described in this section or terminating this Agreement on written notice to Developer.

Bei DaS-V ist diese Prüfungspflicht unter Nr. 1 S.2 bereits mit der Einführung der Anforderungen als Anlage in den Vertrag verknüpft²⁴².

Die vorgenannten Prüfungs- und/oder Rügepflichten auch in zwei der neutralen Muster machen deutlich, welcher Wert darauf gelegt wird, Missverständnisse hinsichtlich der Leistungsbestimmung zu vermeiden.

IV. Besondere Pflichten des Softwarehauses

In vier Mustern werden dem Softwarehaus zusätzliche (ausdrückliche) Pflichten bei der Durchführung des Vertrages auferlegt.

FL fordert vom Softwarehaus die Bereitstellung ausreichender Ressourcen:

1.2. Development Budget and Timetable. Developer shall commit and utilize sufficient resources to meet the milestones and to complete development of the Subject Programs within the development timetable set forth in Exhibit B [omitted] and within the development budget set forth in Exhibit C [omitted]. Developer shall notify Customer of any circumstances, when and as they arise, that may reasonably be anticipated to lead to a material deviation from the development timetable and development milestones set forth in Exhibit B. Developer shall devote sufficient time and effort and shall allocate sufficient personnel resources to the Subject Programs as may be required for development and testing.

Neben dieser allgemeinen Beschreibung wird hinsichtlich der Projektleitung Konkretes gefordert:

1.5. Designation of Supervisor. Developer shall designate an employee who shall be assigned by Developer to supervise the development of the Subject Programs. The employee shall devote substantially all of his or her business time to that endeavor.

Bei AS wird das Softwarehaus zu monatlicher Berichterstattung über den Projektstatus verpflichtet:

²⁴² s.o. 61.

2.7. Progress Reports. Developer agrees to provide Client with monthly progress reports and a statement of Developer's fees and expenses incurred to date. These progress reports are due on the fifteenth (15th) day of each month for the prior month. Each report will contain a description of the current status of the Software, the time spent on Development Services during the month, the tasks on which the time was spent, the estimated progress to be made in the next month, any problems encountered and their proposed solutions, and Developer's estimated ability to meet milestones and other deadlines.

Noch weitergehende Formulierungen findet man bei den Softwarebestellerorientierten Mustern DaS-U und RB-U. So werden bei DaS-U u.a. konkrete Darlegungen zu den im Projekt tätigen Mitarbeitern verlangt:

9. Progress Reports and Meetings

Vendor shall submit monthly written progress reports to User on the first of each month. Each report shall describe the work performed since the preceding report, including but not limited to, the development, testing, and installation status of each function of the system, the names of each employee then working on the project, and a description of the work performed by each listed employee during the report period. The report shall also describe the work expected to be performed during the month following the report. Any uncorrected delays or problems during the report period and their causes shall also be set forth. Progress meetings shall be held in person or by conference call on Monday of each week (unless rescheduled by agreement) and minutes thereof prepared by User and circulated to Vendor.

RB-U sieht zwei Konferenzen mit dazugehöriger Berichtspflicht pro Monat vor:

4.4. Progress Meetings

Twice each month there shall be a meeting to discuss the progress of the projects. At such meetings the Licensor Project Manager shall present a written report to Licensee with respect to project status and progress. Such report shall include a summary of the accomplishments and difficulties during the prior reporting period, and the anticipated results during the next reporting period.

DaS-U bietet für den Projektfortgang nach Erstellung der Spezifikationen (bei DaS-U „Design Document“ genannt) ein unverkennbar zu Lasten des Softwarehauses ausfallendes Verfahren an. Während das Softwarehaus zahlreiche Pflichten treffen, werden dem Softwarebesteller im Rahmen der Projektphasenbeschreibung²⁴³ zusätzliche Lösungsmöglichkeiten vom Vertrag eingeräumt:

²⁴³ S.o. S. 66, Nr. 2. (a) DaS-U.

3. Project Phases

(a) Phase I – System Definition

(i) ...

(ii) *The Design Document shall be presented by Vendor to user for review and comment. Vendor shall modify, supplement, or change the Design Document as requested by User following its review of it. The Design Document shall then approved by User in writing prior to the start of any other work under this agreement. Such approval shall not operate as an acceptance by user of any latent defects, deficiencies, problems, or omissions in the Design Documents or the System.*

(iii) *In the event that Vendor does not provide to User a Design Document which is satisfactory to User, as determined in the sole discretion of User, User may elect to terminate this agreement without further liability or obligation ton Vendor and may retain the Design Document and use it in whatever manner, if any, User sees fit.(...)*

Eine zusätzliche Kündigungsmöglichkeit wird dem Softwarebesteller nach Abschluss der Spezifikationsphase eingeräumt.

25. Termination

(a) ...

(b) *User may, at its sole option, elect to terminate this agreement for convenience, by written notice following the completion of Phase I [Anm.: Gemeint ist die Systemdefinition in Abgrenzung zur „eigentlichen“ Systementwicklungsphase] . In the event of such termination, User shall pay the amount due for Phase I, shall retain and own all rights in and to the Design Document, and shall have no further obligations to Vendor, and the parties shall continue to be bound by the noncompetition and confidentiality sections of this Agreement.*

In RB-U wird besonderer Wert auf die Integration des Softwarebestellers in den Entwicklungsprozess gelegt. Dies wird hier durch konkrete Vorgaben hinsichtlich der personellen Projektbeteiligung zu erreichen versucht. Der Softwarebesteller hat die Pflicht, aber auch das Recht, sich an der Projektdurchführung aktiv zu beteiligen. Geregelt wird also die Form der Zusammenarbeit.

4.1 Licensee Participation

During the entire course of the project, licensee shall provide a project director (the „Project Director“) whose duties shall be to act as liaison between Licensor and Licensee. It is understood that for the System to succeed the Licensee data processing staff should acquire a knowledge of the Systems. As a consequence Licensee shall have the right to have one programmer-analyst in Licensor’s premises. Licensor shall provide desk space for such Licensee employee.

Im Anschluss daran finden sich bei RB-U weitere personalbezogene Vorgaben für das Softwarehaus. Diese erfassen zunächst die geforderte Qualifikation und den Widmungsumfang der vom Softwarehaus eingesetzten Mitarbeiter. Darüber hinaus ermöglichen sie auch die Einflussnahme auf Personalentscheidungen des Softwarehauses.

4.2 Staffing and Staff Qualification

The consulting and professional personnel provided by Licensor hereunder shall have appropriate technical and application skills to enable them to perform their duties hereunder. Licensor shall disclose in the progress reports provided for hereunder the names and experience of Licensor consulting and professional personnel assigned to the project. It is agreed, that (named manager) shall be primarily responsible as the project manager. The project manager shall devote his full time efforts to installation of the System. The project manager shall make himself available if reasonably required and agreed to, to be on-site at the appropriate Licensee site during testing, implementation and other necessary times. Licensor shall provide sufficient staffing to enable the System to be installed in accordance with the implementation schedule. Licensee shall have the right to require Licensor to replace any personell, other than the project manager, working on the project at Licensees premises whom Licensee deems, in its reasonable sole discretion, to be unfit or otherwise unsatisfactory. In the event that (named manager) is no longer associated to Licensor, the responsibilities shall be assumed by an individual with equivalent qualifications, experience, and knowledge of the appropriate System.

DaS-U erweitert diese Möglichkeit zur Einflussnahme um Genehmigungspflichten und Anforderungen an die Art des Beschäftigungsverhältnisses. Zudem ist die Pflicht zum Austausch von Seiten des Softwarebestellers unerwünschtem Personal deutlich weiter gefasst.

11. Project Leader

Vendor agrees to provide the services of _____ to personally supervise all of Vendor's work on the System and to prepare the Design Document. All other personnel assigned by Vendor to work on the System shall be pre-approved by User and shall have necessary technical and application skills and experience and education and shall be full-time employees of Vendor. Vendor shall replace any personnel User finds to be unfit or unsatisfactory for any reason.

In den Softwarebesteller-orientierten Mustern DaS-U und RB-U trifft man schließlich auf sog. „*Most Favored Customer*“-Klauseln. Sie verpflichten das Softwarehaus, für das Projekt vorgesehene Personal bis zum Projektende nicht anderweitig einzusetzen. Dies wird bei RB-U so formuliert:

6.2 Most Favored Customer

For the complete period covered by this Agreement, Licensor agrees to treat Licensee as its most favored customer. In accepting such treatment, Licensor agrees not to reassign any staff assigned to Licensee's projects to other projects until such time the Licensee projects are satisfactorily completed.

Darüber hinaus muss das Softwarehaus dem Softwarebesteller seine Leistungen zu den günstigsten Konditionen anbieten.

Licensor resents that all of the prices, terms, warranties and benefits granted by Licensor hereunder are comparable to or better than the equivalent terms being offered by it to any present customer of Licensor. If during the term of this Agreement Licensor shall enter into arrangements with any other customer providing such customer more favorable terms, this Agreement shall thereupon be deemed amended to provide the same terms to Licensee.

DaS-U geht insofern darüber hinaus, als diese Begünstigungspflicht sich auch auf die Softwarepflege und die Unterstützung sowie neu entwickelte Software bezieht:

24. Most favored User

(...) Vendor shall not charge a price for System maintenance and support which is in excess of that charged to any other customer for comparable services; and Vendor shall inform User of, and make available to User, for license at a reasonable fee not to exceed that charged to any other person, any newly developed software which User may find useful to its business operations.

V. Besondere Pflichten des Softwarebestellers

Besondere Pflichten werden dem Softwarebesteller in den Mustern DaS-V und RB-V auferlegt. Bei DaS-V erfolgt dies im Zusammenhang mit Kündigungspflichten bei der Regelung von verspäteten Leistungen. Wirkt der Softwarebesteller nicht in angemessener Form mit, kann sich das Softwarehaus vom Vertrag lösen bzw. ist bei verspäteter Leistungserbringung entschuldigt:

8. Default

Vendor may terminate the License in the event of a default by customer. Events of default include but are not limited to: (...) (c) Customer's failure to provide reasonable assistance and cooperation to Vendor; (...)

3. Fee

(...) If Vendor is delayed or prevented from performing this Agreement by any cause beyond its reasonable control, including, but not limited to, any failure by customer to provide reasonable assistance and cooperation to Vendor, such a delay shall be excused during the continuance of and to the extent of, that cause, and the period of performance shall be executed to the extent necessary to allow performance after the cause of the delay has been removed.(...)

Bei RB-V findet sich nach einer generellen, beide Vertragsparteien betreffende Verpflichtung zur Kooperation (in Nr. 8) eine detaillierte Auflistung von Handlungserfordernissen auf Seiten des Softwarebestellers (in Nr. 9). Diese bestehen im wesentlichen daraus, alle für erforderlich gehaltenen Informationen zu liefern, die technischen Voraussetzungen für die Softwareentwicklung und Tests zu schaffen sowie ausreichendes Personal bereit zu halten:

8. Cooperation

Both Licensor and Licensee acknowledge and agree that successful development of the Management System so as to become operational in the Licensee's environment shall require their full and mutual good faith cooperation, including, without limitation, the fulfillment by Licensee of the obligations set forth in paragraph 9 below.

9. Licensee Obligations

In addition to providing Licensor with full, good faith cooperation and such information as may be required by Licensor in order to develop and implement the Management System, Licensee shall:

- (a) provide Licensor with specific and detailed information concerning Licensee's work flow, procedures and transaction volumes as they relate to applications of the Management System; provided, however, that the supplying of such information shall not be deemed to alter or supplement in any manner the description of the existing System contained in Schedule „A“ hereto or the description of the changes and modifications necessary to the Existing System in order to develop the Management System, which changes and modifications are specified in Schedule „B“ hereto.;*
- (b) install and maintain at the location referred to in paragraph 3 hereof the hardware and system software specified in Schedule „D“ hereto (the „System Configuration“);*
- (c) make available to each Licensor employee located in Licensee's premises test time on the System Configuration in the amount of at least 30 hours of which shall be between 9:00 a.m. and 5.00 p.m. on business days, which availability shall include sufficient disc storage space to permit Licensor to develop and test the Management System, and access to the*

System Configuration through at least one dedicated terminal for each Licensor employee at the Licensee's premises;
(d) set up, format, and make available to Licensor the frames and tables specified in Schedule „E“ hereto;
(e) make available to Licensor, personnel of Licensee testing the Management System and training users of the Management System;
(f) provide one employee of Licensee who shall have substantial computer systems and project management experience satisfactory to Licensor to act as coordinator of all Licensee activities in connection with the development of the Management System, and to supervise all projects undertaken by Licensee in connection with the modification, preparation, installation and use of the Management System;
(g) convert, or enter into a suitable agreement for Licensor to convert, data files as required to interface with the Management System;
(h) in general, to provide all information and access to the personnel needed to develop and implement the Management System.

VI. Veränderungen (Change Management)

Das Change Management wird in 4 Mustern behandelt. In den Softwarehaus-orientierten Mustern DaS-V und RB-V sowie FL ist es nicht geregelt.

Eine einfache, ziemlich neutrale Form mit dem Schwerpunkt auf gemeinsamer Entscheidung über Änderungen findet sich bei DS. Das Änderungsverfahren kann hier von beiden Vertragspartnern in Form eines Änderungsvorschlages eingeleitet werden. Softwarebesteller und Softwarehaus diskutieren die Auswirkungen auf das Projekt und nehmen ggf. einvernehmlich eine Vertragsergänzung vor. Zum Schutz des Softwarebestellers ist eine Zustimmungspflicht des Softwarehauses vorgesehen, wenn der Veränderungswunsch „reasonable“ und der Softwarebesteller die Kosten zu tragen bereit ist.

1. Software Development Services

(...)

c) Either party may, from time to time during the performance of the development of a particular Product, provide written notice to the other party proposing changes to the applicable Specification or the schedule ("Change Proposal"). Following a Change Proposal, the parties shall promptly meet to determine the financial and schedule impact, if any, and whether, and with what modifications the proposed change is mutually agreeable. No Change Proposal will have any contractually binding effect until formally agreed to in writing by both parties; provided, however, that if XXXXX initiates a Change Proposal, and agrees to pay Developer's additional fees to make such change, Developer may not refuse to accept any reasonably requested Change Proposal.

DaS-U bestimmt, bestellerfreundlicher, dass Änderungen an und Ergänzungen zu den Spezifikationen infolge veränderter Rahmenbedingungen auftreten können, und erlegt dem Softwarehaus für diesen Fall die Pflicht auf, den Softwarebesteller über daraus resultierende Mehrarbeit unter Angabe von Preis- und Terminplanänderungen zu informieren. Die Zusatzarbeiten müssen vom Softwarebesteller autorisiert und dann vom Softwarehaus zu Standardpreisen um 20% reduziert durchgeführt werden.

2. Project Phases

(a) Phase I – System Definition

(...)

(v) The design Document is subject to further supplementation or modification after approval of it necessitated by changing or newly recognized business needs and requirements. If such supplementation or modification requires additional work beyond that originally stated in the approved Design Document, Vendor shall so inform User and promptly provide User with a price for the additional work on a time and materials basis. The time rates shall be the then-current published standard rates charged by Vendor to similarly situated customers, less twenty percent (20 %). The additional work shall not be performed unless the specific work and the price of it are first authorized in writing by a duly authorized officer of User.

Woraus sich Änderungen genau ergeben können, ist im Muster nicht geregelt. Ein Ablehnungsrecht des Softwarehauses hinsichtlich der Durchführung auch umfangreicher Änderungen ist nicht vorgesehen.

AS setzt voraus, dass der Änderungswunsch vom Softwarebesteller ausgeht. Die Anfrage ist vom Softwarehaus innerhalb einer Frist von 10 Tagen auf Kosten, Auswirkungen auf den Terminplan und Ressourcenverfügbarkeit zur Ausführung zu prüfen. Die Durchführung der Änderung kann auch abgelehnt werden. Nimmt das Softwarehaus an, wird der Vertrag auf Grundlage einer Änderungsvereinbarung fortgeschrieben. Sowohl für die Änderungsanfrage als auch für die Änderungsvereinbarung sind bei AS Muster als Anlagen vorgesehen:

5.1.1. Change Order Request. *Client will submit to Developer by means of a written order all requests for additional services that alter, amend, enhance, add to, or delete from the Development Services (the "Change*

Order"). A sample Change Order is set forth in Exhibit B, attached to and made part of this Agreement.

5.1.2. Change Order Evaluation. Developer will evaluate each Change Order, without charge, within ten (10) days of its receipt and will provide Client with: (a) the change in cost as a result of the Change Order; (b) the impact, if any, of the Change Order on the completion date and milestone deadlines; and (c) the availability of Developer's resources to carry out the Change Order. Developer may, in its sole discretion, refuse to carry out any proposed Change Order.

5.1.3. Change Order Acceptance. If Developer agrees to carry out the proposed Change Order, Client will promptly give Developer written authorization to carry out the Change Order by executing the Change Order in the manner provided on Exhibit B.

5.2. Performance of Change Order. Developer will begin performance in accordance with a Change Order immediately upon its execution by Client. Developer has no obligation to perform any additional services before receiving the executed Change Order and Client has no obligation to pay Developer for any services performed pursuant to a Change Order before that Change Order is executed by Client.

5.3. Change Order as Binding Agreement. Once fully completed and executed, each Change Order is deemed to be incorporated into and to be part of this Agreement and will constitute a formal amendment to this Agreement, adjusting fees, and completion dates to the extent set forth in the Change Order.

RB-U unterscheidet zwischen kleineren („*minor*“) Modifikationen und Änderungen nach Annahme der Spezifikationen (und vor der ersten (Teil-) Abnahme). Kleine Änderungen müssen dokumentiert werden, bedürfen der Zustimmung des Softwarebestellers und dürfen keine Auswirkungen auf den Preis haben.

RB-U geht ebenfalls davon aus, dass ein Änderungsverlangen durch den Softwarebesteller initiiert wird. Das Softwarehaus hat innerhalb von 20 Tagen zu Kosten- und Terminfolgen Stellung zu nehmen. Das Änderungsverlangen erlangt Wirksamkeit, wenn der Softwarebesteller es innerhalb von weiteren 20 Tagen bestätigt.

4.13 Changes After Detailed Design

A. Licensor in developing the Software System may make minor modifications to the Software System and Detailed Design Specifications if such minor modifications do not limit, diminish or affect the functional operation or use of the System or the output, or result in the System failing to comply with the volume and response time criteria contained in this agreement. All such changes shall be documented in writing, a copy of which shall be submitted to Licensee for prior approval. There shall be no

change in the prices set forth in this Agreement as a result of such minor specifications.

B. After acceptance of the detailed Design Specifications, and before Preliminary Acceptance of any Application, Licensee shall have the right to request from Licensor in writing a change to the Software System or Detailed Design Specifications.

If a change is requested, Licensor shall, within a reasonable time not to exceed twenty days inform Licensee in writing if such change would result in an extension of the Implementation Schedule, or additional cost to Licensee, giving the details thereof. Licensor shall use the best efforts to respond as expeditiously as possible. A change requested by Licensee requiring no extension or additional cost shall be considered effective if within twenty days Licensee consents in writing thereto. A change for which an extension or increase has been specified by Licensor shall be considered effective if within twenty days thereafter Licensee consents in writing to such extension or increase.

VII. Zusammenhang zwischen Leistungsbestimmung und Einstandspflicht

Bemisst man die Anforderungen an die zu erbringende Leistung an den Gewährleistungsregeln, so ergeben sich bemerkenswerte Unterschiede. Bei AS wird die strikte Übereinstimmung mit dem „Scope of work“ verlangt:

*8.1.1. **Compliance with Specifications.** Developer warrants that the Development Services and the Software will strictly comply with the descriptions (including performance capabilities, completeness, specifications, configurations, and function) that appear in the Scope of Work or in any Change Orders.*

FL und DS verwenden demgegenüber abgeschwächte Formulierungen. Während bei DS eine akzeptable Implementation der Spezifikationen und ein einjähriges Funktionieren gemäß der Spezifikationen verlangt wird:

4. Warranty.

(a) Developer warrants that the Specifications will represent an acceptable implementation of the Functional Requirements. Developer warrants that for a period of twelve (12) months after XXXXX's first commercial shipment of the Product excluding shipments for beta testing, (the "Warranty Period"), the Product will perform in accordance with the Specifications.(...)

sieht FL vor, die Software müsse - ebenfalls für ein Jahr - bezüglich aller wesentlicher Aspekte mit den Spezifikationen und den

Ausführungsstandards übereinstimmen, wobei nachträgliche Änderungen durch den Softwarebesteller ausgenommen sind:

*6.01. **Limited Warranty.** Developer warrants, for the benefit only of Customer, that the Subject Programs shall conform in all material respects to the Specifications and Performance Standards (except for subsequent modifications made at Customer's request).*

Die übrigen Muster fallen deutlich unterschiedlich aus, je nachdem, ob sie aus Sicht des Softwarehauses (DaS-V und RB-V) oder des Softwarebestellers (DaS-U und RB-U) formuliert sind.

Die erstgenannten verwenden übereinstimmend den Maßstab der substantiellen Übereinstimmung:

DaS-V unter 6. Warranty and Remedies:

„(...) the software will perform substantially in accordance with the specifications for the software (...))

RB-V unter 13. Warranty:

„ (...) the Management System will substantially conform to the specifications contained in Schedule „A“, as modified by the Modifications listed in Schedule „B“ (...))

RB-U verlangt demgegenüber ohne Beschränkung, dass die Software mit den vereinbarten Anforderungen übereinstimmt:

5.5. System Requirements

(...) such applicaton shall comply with Licensee´s requirements as disclosed in the Detailed Design Specifications. (...)

und dass auch die Antwortzeiten des Systems sich innerhalb des vereinbarten Rahmens bewegen²⁴⁴.

Noch weitergehend verlangt DaS-U, dass die Software die Bedürfnisse des Softwarebestellers, wie sie in den Anforderungsbestimmungen formuliert sind, erfüllt.

17. Warranties

(a) Warranty: Vendor, as an expert in the field warrants that: the system shall meet User´s needs (as defined in the approved Design Document and

²⁴⁴ Einzelheiten bei RB-U unter 5.6 *Terminal Response Time*.

in Exhibit „A“) and shall be substantially free from programming errors and defects in workmanship and materials.(...) ²⁴⁵

Hinsichtlich der Preisgestaltung findet sich in allen Mustern eine Fixpreisvereinbarung²⁴⁶. Bei den tendenziell ausgerichteten Mustern sind Zusatzleistungen entweder gesondert zu vergüten²⁴⁷ oder im Gesamtpreis inbegriffen²⁴⁸.

B. Beispiele aus der Praxis

Als Beispiele aus der Praxis wurden drei Verträge ausgewählt²⁴⁹, die Softwareentwicklungsprojekte in den USA betreffen²⁵⁰. Der erste wurde zwischen einem Softwarehaus aus Nebraska - **First Data Resources** - und einem Softwarebesteller aus New York - **Prodigy Services** - (im folgenden FP)²⁵¹, der zweite zwischen einem Softwarehaus in Colorado - **Evolving Systems** - und der **American Telephone and Telegraph Company** in New Jersey (AE)²⁵², der dritte zwischen einem Softwarehaus in Manitoba/Canada

²⁴⁵ Auch hier sind weitere Einzelheiten wie eine *Response Time warranty* vorgesehen.

²⁴⁶ Bei DL unter 3 (a); bei FL unter 3.01; Bei AS, unter 3.1, wobei eine Basiszahlung (*Base development Fee*) und Zusatzzahlungen (*Additional Development Fees*) für Änderungen vorgesehen sind.

²⁴⁷ So bei RB-V nach Nr. 11 a.E. „*general consulting services*“.

²⁴⁸ So bei DaS-U unter 5., das nur „*reasonable travel expenses incurred in travelling to User facilities*“ von dem Fixpreis ausnimmt.

²⁴⁹ Die Auswahl spiegelt unterschiedliche Ausgangssituationen wider, sowohl was die Gesamttendenz des Vertrages (mehr an den Interessen des Softwarebestellers (so in AE, siehe sogleich) bzw. des Softwarehauses (so in FP) orientiert), als auch was die faktische Grundlage (Neuentwicklung in OD, Weiterentwicklung in FP) anbelangt.

²⁵⁰ An dieser Stelle möchte ich nicht versäumen, mich bei dem Kollegen, Prof. Anuj Desai von der University of Wisconsin, Madison zu bedanken, der mich bei der Suche nach Material sehr hilfreich unterstützt hat. Die Verträge stammen aus der sehr umfangreichen Datenbank EDGAR der US Securities and Exchange Commission (<http://www.sec.gov/edgar.shtml>), sowie der Datenbank CORI des Contracting and Organizations Research Institute der University of Missouri, Columbia (cori.missouri.edu/search).

²⁵¹ Vertrag zwischen First Data Resources Inc. und Prodigy Services Company, unterzeichnet im Dezember 1992 und die (Weiter-) Entwicklung eines Buchungs-, Rechnungs- und Mitgliederverwaltungssystems betreffend (EDGAR: Central Index Key No. 0000824740, File No.: 0000950109-98-004594). Die Parteien hatten im Mai 1986 einen Vertrag über die Entwicklung eines Subscription Management Systems („SMS“) abgeschlossen. Dieses System sollte unter dem hier untersuchten Vertrag weiterentwickelt und um ein Member Administrative System („MAS“) ergänzt werden. Das Softwarehaus wird im Vertragstext „FDRI“, der Softwarebesteller „Prodigy“ genannt.

²⁵² Rahmenvertrag zwischen der American Telephone and Telegraph Company und der Evolving Systems Inc. aus dem Jahr 1998 (EDGAR: Central Index Key 0001052054, File No. 333-43973). Das Softwarehaus wird im Vertragstext als „Supplier“, der Softwarebesteller als „Company“ bezeichnet.

- OMT-Technologies - und einem Unternehmen in Los Angeles/ California
- DMX Incorporation - (OD)²⁵³ geschlossen.

I. Vertragsart

OD besteht aus zwei Teilen, einem Purchase Agreement und einem Software License Agreement, die durch verschiedenen Anhänge ergänzt werden, wobei hier ein sog. Rider von Bedeutung ist. Die Bezeichnung als Purchase kann dabei insofern als irreführend angesehen werden, als bereits unter Nr. 2. davon die Rede ist, das Softwarehaus habe Software, Services und Support zu liefern²⁵⁴. Auch enthält der Lizenzvertrag keine über die in den anderen Verträgen ebenfalls enthaltenen diesbezüglichen Regelungen wesentlich hinausgehenden Bestimmungen.

FP ist als Software Development and Processing Services Agreement bezeichnet und mit zahlreichen Anhängen versehen, während AE den Titel Software Development Agreement trägt und sich aus drei Teilen, der unbenannten Section I, der mit General Provisions of a Software Development Contract bezeichneten Section II²⁵⁵, sowie sog. Authorization Letters zusammensetzt. Sowohl bei OD²⁵⁶ als auch bei FP findet sich zudem die Bestimmung, dass Überschriften keine Bedeutung beizumessen sei²⁵⁷:

Headings. The title or headings of various paragraphs, Sections and Articles hereof are intended solely for convenience of reference and shall not be construed for any purpose whatsoever to modify or explain or place any construction upon any of the provisions of this Agreement.

Sowohl Aufbau als auch Bezeichnung der Verträge eignen sich nicht als Grundlage für Schlussfolgerungen hinsichtlich der Leistungsbestimmung.

²⁵³ Vertrag zwischen OMT Technologies Inc. aus Winnipeg/Manitoba in Kanada und der DMX Inc. aus Los Angeles, Kalifornien über Software-, Service- und Supportleistungen, unterzeichnet im März 1998 (EDGAR, Central Index Key 0001040449, File No. 00022815 v., CORI, File No. 1040449-98-001350b.html). Dass es sich bei dem Softwarehaus um ein kanadisches Unternehmen handelt, kann hier vernachlässigt werden, da der Vertrag dem kalifornischen Recht unterstellt wurde (siehe Nr. 28 des Vertrages), zumal das kanadische Rechtssystem mit Ausnahme des Bundesstaates Quebec – wie das der USA – auf dem englischen Common Law basiert. Das Softwarehaus wird im Vertragstext als „Vendor“, der Softwarebesteller als „Customer“ bezeichnet.

²⁵⁴ Siehe sogleich S. 82.

²⁵⁵ Bemerkenswert ist dabei, dass die Gliederung nach alphabetisch angeordneten Themen, von A wie „Acceptance“ bis W wie „Work done by others“ angeordnet ist.

²⁵⁶ Unter Nr. 22.

²⁵⁷ Hier wird die extensive Formulierung aus FP, Nr. 9.6 wiedergegeben.

Während bei AE eine exklusive Lizenz des Softwarebestellers vorgesehen ist²⁵⁸, findet sich bei OD nur das Einräumen einer einfachen, nicht übertragbaren Lizenz, deren Übergang zudem von der Abnahme und der vertragsgemäßen Zahlung abhängig sein soll²⁵⁹. In FP wird ausdrücklich darauf hingewiesen, dass sämtliche Rechte an den vom Softwarehaus entwickelten Systemen ausschließlich bei diesem verbleiben²⁶⁰. Hat der Softwarebesteller eigene *processes* entwickelt, kann er das Softwarehaus darüber informieren, dass er diesbezüglich eine Ausnahme als gegeben ansieht und eigene Rechte beansprucht²⁶¹.

Aus den Vertragstexten ergibt sich, dass *Service* entweder als die zentrale Leistung (FP und AE) oder jedenfalls als wesentlicher Vertragsbestandteil anzusehen ist. So heißt es in FP in der Präambel:

“WHEREAS, PRODIGY desires to engage FDRI to continue to provide full life cycle systems development, data processing services and other services for PRODIGY in connection with PRODIGY’S member billing operations and to create a software development team at FDRI dedicated to the development of a Member Administrative System (the “MAS System”) for PRODIGY (...)”

In AE trifft man ebenfalls in der Präambel auf folgenden Wortlaut,

WHEREAS, Company desires to retain Supplier to furnish the programming, training, consulting or other services (the “Services”) (...)

²⁵⁸ Unter *“Software, Titel and Rights”*.

²⁵⁹ Purchase Agreement Nr. 4.

²⁶⁰ Nr. 1.1.8.2 (a), dem Softwarehaus wird dabei allerdings untersagt, - während der Vertragslaufzeit - Rechte an Wettbewerber zu übertragen. Die Lizenzrechte sind an den bestehenden Vertrag geknüpft - Nr. 4.1.1 und 4.1.2 (!) -, wobei der Softwarebesteller die Möglichkeit hat, bei Vertragsende das Recht auf zeitlich unbeschränkte Nutzung zu erwerben (4.1.3). Macht der Besteller von dieser Möglichkeit Gebrauch, dann tritt ein fünfjähriges Verbot des Verkaufes an Wettbewerber in Kraft (4.1.3 (b) iii), und das Softwarehaus muss bei der lückenlosen Übertragung - was einer vollständigen Übertragung des Know-Hows ohne Unterbrechung des produktiven Einsatzes der Software entspricht - mitwirken (4.1.3 (d)).

²⁶¹ 1.1.8.2 (b): Wenn das Softwarehaus dieser Einordnung widerspricht, wird ein spezielles Einigungsverfahren in Gang gesetzt, das - wenn sich die Parteien nicht einigen können - in ein schiedsgerichtliches Verfahren mündet.

der unter *Statement of Work* wie folgt wiederholt und ergänzt wird:

STATEMENT OF WORK. Supplier shall provide the Services described in the Authorization Letter and deliver the Pre-existing Software and the Developed Software (collectively referred to as "Software") and all other Deliverables described in the Authorization Letter to Company at Company's location in accordance with the Work schedule set forth in the Authorization Letter.

Bei OD werden drei Leistungsbestandteile unterschieden:

TRANSACTION AND SCOPE. Vendor agrees to provide to Customer and Customer agrees to acquire from Vendor the following:

- a) SOFTWARE. The software listed in Schedule C.*
- b) SERVICES. Services as listed in Schedule D.*
- c) SUPPORT. Support as listed in Schedule E.*

In dem an den Interessen des Softwarebestellers ausgerichteten Muster AE wird neben dem *Service* die Lieferung der Software zur ausdrücklichen Pflicht gemacht. Während man eine dementsprechende Regelung bei FP vergeblich sucht, ergänzt OD im Anhang (*rider*) unter Nr. 1 folgende Formulierung:

1. To fulfill the requirements of SCHEDULE A attached to the Agreement, Vendor recommends the software manufactured, developed or supplied by Vendor and other components to be supplied by Vendor (...)

Das Softwarehaus „empfiehlt“ für eine Erfüllung der Leistungsanforderungen den Einsatz nicht näher definierter anderer, vom Softwarehaus zu liefernder Komponenten. Dies erschwert dem Softwarebesteller den Einsatz von Konkurrenzprodukten, da er sich damit ggf. dem Einwand des Softwarehauses aussetzt, ein den Anforderungen entsprechendes Ergebnis sei nur mit dessen Produkten zu erzielen.

II. Bestandteile der Lieferung

Hierzu findet sich bei OD die Pflicht des Softwarehauses, *operating manuals* und *complete source code listings* zu liefern²⁶². In der Anlage A²⁶³ wird als *Scope of Project*

(...) to provide software, consulting and professional services for 120 automated music formats using MPEG I Layer 2 WAV audio compression.

genannt.

Bei AE wird der Lieferungsumfang über die *Pre-existing Software* und die *Developed Software* hinaus auf *Deliverables* erstreckt²⁶⁴.

FP stellt die Bereitstellung weitgehend definierter Teams in den Vordergrund. Dies gilt sowohl für die Anpassungsleistungen,

1.1.7 SMS Enhancements and Maintenance Team. As of the date of this Agreement, FDRI employs certain individuals (referred to in Exhibit 6 as the "SMS Enhancements and Maintenance Team") who are and/or shall be assigned exclusively to work on enhancements and maintenance for the SMS Product in the following capacities:

<i>Personnel</i>	<i>Rates/Per Day/Per Person</i>
<i>One Team Leader (Development Manager)</i>	<i>[**]</i>
<i>Two Systems Analysts</i>	<i>[**]</i>
<i>Six Programmers</i>	<i>[**]</i>
<i>One Technical Support Person</i>	<i>[**]</i>
<i>One Product Integrity (Testing/Quality Assurance) Person</i>	<i>[**]</i>

als auch für die Neuprogrammierung:

1.1.8.1 MAS System Development. In consideration of the financial and technical resources committed hereunder by PRODIGY for the development/enhancement of SMS, FDRI agrees to provide for the development of MAS a level of support equivalent to that identified in Section 1.1.7 above. Therefore, except where otherwise required or permitted under this Section 1.1.8.1, FDRI shall employ, in the requisite positions and at the requisite skill levels, eleven (11) individuals who will be

²⁶² Rider Nr. 9.

²⁶³ Schedule A – Vendor Specification.

²⁶⁴ Siehe soeben im *Statement of Work* bei AE, S. 81f.

dedicated by FDRI to the development of the MAS System (the "MAS Development Team").

III. Spezifikationen

Alle Verträge beziehen sich auf technische Anlagen, in denen das Projekt näher beschrieben wird. Am kürzesten fällt diese Bezugnahme bei AE unter „*Statement of Work*“ aus:

The Software and such other Deliverables shall perform and comply with the specifications set forth in the Authorization Letter (the "Specifications").

Die *Authorization Letters* sollen dabei ausweislich der Präambel sukzessive erstellt und gemeinsam umgesetzt werden:

*WHEREAS,
Company desires to retain Supplier to furnish the programming, training, consulting or other services (the "Services") specified in Work Authorization Letters (each an "Authorization Letter") to be issued from time to time and executed by both parties;*

Wer die Spezifikationen also vornimmt, ist zumindest im Vertrag nicht ausdrücklich geregelt²⁶⁵.

Dass die Spezifikationen sich erst im Verlaufe des Projektes herauskristallisieren, wird bei FP noch deutlicher. Dies gilt in abgeschwächter Form zunächst für die weiter zu entwickelnde Software, bei der eine gemeinsame Prioritätenbildung erfolgen soll²⁶⁶:

Each SMS System Enhancement and/or Maintenance Change shall be jointly prioritized by FDRI and PRODIGY, and implemented in accordance with each SMS Joint Detailed Project Plan that results from Exhibit 1 Section 2 and with the SMS Implementation Plans that result from Exhibit 5 Section 5.

²⁶⁵ Aus dem Gesamtzusammenhang des Vertrages kann die Vermutung abgeleitet werden, der Softwarebesteller sei für die Vorgaben verantwortlich. Dafür spricht der Wortlaut („issued“), insbesondere angesichts der Tatsache, dass es sich bei dem Besteller um ein sehr großes Unternehmen mit einer eigenen IT-Abteilung handelt. Auch die Terminologie der „Authorization Letters“ lässt es nahe liegend erscheinen, dass das Softwarehaus zur Ausführung vom Softwarebesteller vorgegebener Arbeiten autorisiert wird, wenngleich dieser Schluss nicht zwingend ist.

²⁶⁶ Unter 1.1.1.1.

Es wird deutlicher hinsichtlich der neu zu entwickelnden Software. Hier findet sich zunächst eine der soeben aufgeführten ähnliche Formulierung²⁶⁷:

MAS shall be developed, tested and implemented in modules and subsystems, which shall be jointly prioritized by FDRI and PRODIGY, and implemented in accordance with each MAS Joint Detailed Project Plan that results from Exhibit 1 Section 2 and with the MAS Implementation Plans that result from Exhibit 5 Section 6.

Diese wird im Anschluss daran weiter präzisiert²⁶⁸:

The Project Managers shall be responsible for the day-to-day management and successful completion of the MAS and SMS projects (as set forth in Exhibit 1) and of the Joint Detailed Project Plans required under Exhibit 1 Section 2. The Project Managers shall also establish a MAS Design Baseline that shall be mutually agreed to by FDRI and PRODIGY, having been reviewed and approved by appropriate management. The MAS Design Baseline shall consist of the Functional and Data Specifications from the respective Business Area Analysis Units designated by the parties during the Logical Phase of the MAS system development life cycle. The Logical Phase work product deliverables that will define the MAS Design Baseline will be: Process and Data Definitions; Process Decomposition Diagrams (Logical Process Model); Data Flow Diagrams; Entity Relationship Diagrams (Logical Entity Model); and Association Matrices.

Die Gemeinschaftlichkeit der Entwicklungsarbeit ergibt sich hier aus mehreren Gesichtspunkten. Zum einen wird hier als selbstverständlich vorausgesetzt, dass die Spezifikation von Daten und Funktionen von beiden Vertragsparteien stammt („*designated by the parties*“). Weiterhin soll eine Entwurfsrichtlinie gemeinsam entwickelt und vom Management geprüft und bestätigt werden. Schließlich ist die gemeinsame Projektleitung, also eine Verknüpfung auch auf personeller Ebene vorgesehen²⁶⁹.

OD beschreitet einen anderen Weg. Hier sind die Spezifikationen in einem detailliert angegebenen Verfahren bereits bei Vertragsabschluss festgelegt. Der Softwarebesteller hat seine Wünsche und Bedürfnisse geäußert, das Softwarehaus hat diese Vorgaben evaluiert und in eine detaillierte

²⁶⁷ Unter 1.1.1.2.

²⁶⁸ Unter 1.1.2.

²⁶⁹ 1.1.2 Satz 1 lautet: *FDRI and PRODIGY agree to assign and make available one Project Manager each and other key personnel for the duration and completion of the MAS systems development, MAS maintenance, SMS operations, SMS maintenance and MAS operations, under the direction of the Project Managers to ensure the completion of the SMS and MAS projects as specified in the Joint Detailed Project Plan.*

Spezifikation umgesetzt, die der Softwarebesteller im Anschluss daran seinerseits auf eine akkurate Abbildung der Vorgaben hin prüft.

5. *NEEDS AND REQUIREMENTS. The parties hereto agree that:*
- a) *Customer has communicated its needs and requirements to Vendor and has indicated what, if any, unusual conditions or uses needed to be taken into account by Vendor in preparing its proposal;*
 - b) *Vendor has evaluated the information supplied by Customer and has provided Customer with a detailed Vendor Specification (Schedule A) setting forth Vendor's understanding of Customer's needs and requirements as well as the items (including detailed specifications) which will satisfy the needs and requirements contained therein;*
 - c) *The specifications set forth in the Vendor Specification shall be deemed to constitute the maximum specifications to be met by the items listed therein;*
 - d) *Customer has carefully reviewed the Vendor Specification (Schedule A) and is satisfied that its needs and requirements are accurately and completely described, and that the items contained in Schedule A will satisfy these needs and requirements so long as the listed specifications are met. Any errors, omissions of ambiguities are not to be deemed to result from the objectives set out by the Customer and shall be brought to the Customer's attention by the Vendor for consideration.*

Bemerkenswert ist der letzte Satz, wonach Fehler, Auslassungen oder Ungenauigkeiten nicht der unzureichenden Vorgabe des Softwarebestellers angelastet werden dürfen, sondern diesbezüglich eine Aufklärungspflicht von Seiten des Softwarehauses konstituiert wird. Der Softwarebesteller ist also einerseits gehalten, möglichst genau zu definieren, was er will, und den Entwurf des Softwarehauses auch auf die Wiedergabe seiner Intentionen zu prüfen. Das Softwarehaus wird auf der anderen Seite jedoch nicht aus der Pflicht entlassen, auf unzureichende oder fehlerhafte Vorgaben des Bestellers aufmerksam zu machen.

Ergänzt wird diese Regelung durch eine Generalklausel im Anhang, wonach alle seitens des Softwarebestellers bekannt gemachten Spezifikationen durch die Software abgedeckt werden müssen²⁷⁰:

2. *For all purposes of the Agreement, all references to the "System" shall be deemed to include all components, equipment, program products, systems, services, training and documentation referenced in the Agreement or necessary to enable the System to operate in accordance with all Vendors' published specifications with respect thereto, SCHEDULE A and all of the terms and conditions of the Agreement*

²⁷⁰ Rider Nr. 2.

(collectively, the "Performance Standards"). Vendor agrees to provide the necessary skills and services to design, develop, test, document and install the Custom Portions and other necessary design and implementation services so that the entire System meets the "Performance Standards."

IV. Besondere Pflichten des Softwarehauses

Bei FP findet sich zunächst die bereits benannte präzise Anforderung an das vom Softwarehaus bereitzustellende Projektteam. Die Bedeutung dieser Regelung wird durch eine ergänzende Bestimmung zu den *Performance Standards* unterstrichen, die hier auszugsweise wiedergegeben wird²⁷¹:

PERFORMANCE STANDARDS - SMS and MAS

3.1.1 FDRl Responsibilities. FDRl shall maintain the computer capacity and staff necessary to provide services in accordance with the performance standards set forth in this Article 3

(NOTE: Unless otherwise stated, all attainment measurements to be calculated based on calendar days in a calendar month):

*a) As measured by the FDRl host computer, on-line system downtime shall not exceed [**] during the period from 6:00 a.m. to 3:00 a.m. Central Time for any calendar month. For purposes of this item (a), downtime does not include telecommunications lines or individual terminals, controllers or modems on PRODIGY property. (Critical)*
(NOTE: for the purposes of the provisions of this Section 3.1.1, "on-line system" shall mean a system in which the input data enters the computer directly from the point of origin (e.g., computer terminal or workstation) and the output is transmitted directly to its ultimate destination (e.g., computer terminal or workstation) without any intermediate stops or manual intervention.

*(b) The FDRl on-line system shall not experience more than [**] downtime occurrences during a calendar month. This downtime does not include FDRl nightly downtime to update the data base or any scheduled off-hours maintenance. (Critical) (...)*

Diese Regelung enthält zunächst die allgemeine Forderung hinsichtlich ausreichender personeller Ausstattung und im Anschluss daran konkrete Leistungsdaten hinsichtlich der Systemverfügbarkeit, ausgedrückt in Ausfalltoleranz bzgl. Dauer (3.1.1 (a)) und Häufigkeit (3.1.1 (b)). Der in Klammern gesetzt Hinweis (Critical) ist für die Konsequenz im Hinblick auf Nichterfüllung von Bedeutung. Werden drei kritische Anforderungen nicht

²⁷¹ Nr. 3.1., vergleichbare Regelungen werden in der IT-Branche häufig als sog. „Service-Level“ Vereinbarungen bezeichnet.

erfüllt, berechtigt dies den Softwarebesteller zu fristloser Vertragskündigung²⁷².

Über die Einhaltung der *Performance Standards* hat das Softwarehaus monatlich zu berichten. Ergeben sich dabei Abweichungen, so hat dies unmittelbare finanzielle Folgen in Form von Abzügen auf ausstehende Zahlungen²⁷³:

SMS and MAS Performance Reporting and Remedies.

(a) During each calendar month, FDRI agrees to provide PRODIGY with a monthly report listing the performance standards set forth in this Article 3 and FDRI's performance during the just concluded calendar month relative to those performance standards. Within fifteen (15) business days after the end of said calendar month, FDRI shall then review this report with PRODIGY. Included in each such report shall be an analysis of the data used to derive FDRI's performance experience.

*(b) If FDRI fails [**] to perform in accordance with any 3 of the performance standards set forth in Section 3.1.1 and not identified as "critical," FDRI shall credit PRODIGY with a [**] reduction on its bill for processing services during the [**] such month. If such failure continues with respect to such standards for a [**] such month, FDRI shall credit PRODIGY with a [**] reduction on its bill for processing services during such [**] month. If such failure continues with respect to such standards for a [**] such month, FDRI shall credit PRODIGY with a [**] reduction on its bill for processing services during such [**] month. If such failure continues with respect to such standards for a [**] such month, FDRI shall credit PRODIGY with a [**] reduction on its bill for processing services during such [**] month and on the processing services bills for any and all consecutive, succeeding months in which such failure continues.*

*(c) If FDRI fails [**] to perform in accordance with any 3 of the performance standards set forth in Section 3.1.1 and identified as "Critical", FDRI shall credit PRODIGY with a [**] reduction on its bill for processing services during the [**] such month. Continuous failure in any and all consecutive, succeeding months shall result in a [**] reduction on the applicable bill(s) for processing services until the failure is rectified.*

(d) FDRI's failure to meet a performance standard due to any cause beyond its reasonable control and not unique to FDRI such as, but not limited to, the malfunction or failure of any entity from which FDRI must

²⁷² Die allerdings verschiedene weitere Voraussetzungen hat:

2.2.2.2 PRODIGY may terminate this Agreement immediately upon written notice to FDRI in the event that: (i) in accordance with Section 3.1.1(c) FDRI's failure perform in accordance with any 3 of the performance standards identified as "Critical" in Section 3.1.1 results in a 50% reduction in PRODIGY'S processing services bills; and (ii) such 50% reduction remains in effect for nine or more consecutive calendar months. If this Agreement is terminated pursuant to this Section 2.2.2.2, PRODIGY's sole and total financial liability to FDRI shall be as set forth in Section 2.2.3 below.

²⁷³ Unter 3.1.2.

obtain information or have electronic contact in order to perform the services set forth in this Agreement shall not be considered to be a failure for which FDRI shall be responsible under this Agreement.

(e) PRODIGY agrees that due to the difficulty of determining and calculating its damages upon FDRI's failure to perform in accordance with the performance standards included in this Article 3, the remedies set forth in this Article 3 and Section 2.2.2.2 of this Agreement shall be PRODIGY's sole and exclusive remedies, and PRODIGY hereby elects to waive any and all other remedies to which PRODIGY may be entitled under this Agreement, for any claim based on FDRI's failure to perform in accordance with such performance standards.

Die Einstandspflicht des Softwarehauses wird unter (d) im Hinblick auf Ereignisse begrenzt, die außerhalb seiner Einflussosphäre liegen²⁷⁴.

Interessant ist die den Haftungsumfang erklärende und begrenzende Regelung unter (e). Dies verleiht den im Vertrag fixierten Anforderungen an die Leistungserbringung zusätzliches Gewicht²⁷⁵.

Ein Schwerpunkt bei OD liegt auf termingerechter Vertragserfüllung. Wird diese nicht erbracht, fallen beträchtliche Abzüge hinsichtlich ausstehender oder erbrachter Zahlungen bzw. Vertragsstrafen an. Diese treten, anders als bei FP, auch neben weitere mögliche Ansprüche²⁷⁶:

12. *In addition to any other remedies available to Customer hereunder, the amounts payable by Customer to Vendor pursuant to the Agreement shall be reduced by three thousand dollars (\$3,000.00) for each day that Final Acceptance is delayed beyond its final date for completion in the implementation schedule (SCHEDULE G).*
13. *Upon a material breach by Vendor of its obligations, and after notice to Vendor of such breach and a reasonable opportunity to cure such breach, Customer may, at its option to be exercised in its sole discretion and in addition to any other rights or remedies to which Customer may be entitled hereunder or at law or in equity:*
 - (a) *Require that Vendor return all amounts previously paid by Customer to Vendor hereunder within five (5) days after notice to Vendor from Customer demanding such payment, and upon such payment, title and risk of loss to the System shall pass to Vendor; or.*
 - (b) *Upon notice to Vendor, terminate further performance by Vendor of all or any portion of Vendors obligations hereunder and, at Vendor's expense, seek assistance from other persons or entities in completing performance of such obligations.*

²⁷⁴ Diese Klausel ist nicht als Force Majeure Regelung zu verstehen. Force Majeure ist im Vertrag gesondert unter 5.3. geregelt.

²⁷⁵ Im Vertrag finden sich an anderer Stelle Regelungen zu Gewährleistung und Haftung (5. und 6.) Diese sind allerdings mit den üblichen weitgehenden Ausschlussbestimmungen (*Disclaimer*) versehen (5.1. a.E.; 6.4) und enthalten auch eine Haftungsregelung des Softwarebestellers bei eigener Fahrlässigkeit (5.2).

²⁷⁶ Im *Rider* unter 12. und 14., bei Nr. 13 zeigt sich einmal mehr, dass es im amerikanischen Recht am einer den §§ 631ff vergleichbaren gesetzlichen (Auffang-) Regelung fehlt.

14. *For each calendar day after any due date that Vendor has failed to achieve a task or deliverable, Customer, at its option, shall be entitled to the following payments or credits ("Delay Credit"): (i) for the first fifteen (15) calendar days after each such due date, the Delay Credit shall be Five Hundred Dollars (\$500.00) per calendar day; (ii) for the next fifteen (15) calendar days after the unmet due date, the Delay Credit shall be One Thousand Dollars (\$1,000.00) per calendar day; and (iii) thereafter, the Delay Credit shall be Two Thousand Dollars (\$2,000.00) per calendar day.*

Bei OD findet sich zudem eine Begünstigungsregel, wonach das Softwarehaus dem Softwarebesteller die günstigsten Konditionen anbieten muss²⁷⁷:

The remedial rates charged to Customer shall be the lowest such rates charged any Customer of Vendor for comparable service. The rates may be increased annually but to no more than the lesser of (a) the lowest rates then being charged any Customer of Vendor for comparable service, or (b) five percent (5%) of the previous year's rates. No additional charges shall be applied for service unless the service and charges are approved in writing by a duly authorized representative of Customer. Such charges must be identified in the contract or in a duly executed written change order. Vendor must warrant that all prices and charges are the lowest it offers any Customer for comparable products and services.

AE verpflichtet das Softwarehaus neben konkreten vertraglichen Obligationen zur Einhaltung allgemeiner Standards²⁷⁸:

Supplier represents to Company and its customers that the Work will be performed in a worker like manner in accordance with professional standards in the field and subject to the acceptance criteria established in this Agreement or the Authorization Letter.

Die starke Position des Softwarebestellers in AE zeigt sich u.a. in dessen Berechtigung, einen Austausch von Mitarbeitern verlangen zu können²⁷⁹:

Company has the right at any time for reasonable cause (prior to and after assignment to Company's Work) to reject or have Supplier remove Supplier's employee(s) from the Work under this Agreement upon notice to Supplier. Upon such notice, Supplier shall, at company's request, replace the Supplier employee(s). In the event of any staffing change, the Company shall not be charged for the time required to train the replacement. The amount of noncompensatory training time, if any, shall be mutually determined by Supplier and Company's Representative.

²⁷⁷ Im Rider unter 5.

²⁷⁸ Unter *Representation* in Section II.

²⁷⁹ Unter *Supplier Employees* in Section II.

V. Besondere Pflichten des Softwarebestellers

Besondere Pflichten des Softwarebestellers finden sich bei AD zunächst hinsichtlich der Arbeitsumgebung (angemessene Ausstattung der Arbeitsplätze des Softwarehauses beim Softwarebesteller hinsichtlich Zuleitungen, Möbel etc.). Für die zeitgerechte Ausstattung, die allerdings nicht die Hardware umfasst, ist der Softwarebesteller verantwortlich²⁸⁰:

CUSTOMER RESPONSIBILITIES. Customer shall be responsible for timely site preparation including, but not limited to, the provision of adequate electrical power and sufficient number and type of electrical outlets, dust and smoke control provisions, adequate furniture, and sufficient work space for Vendor's personnel to perform installation. Customer shall be responsible for equipment cabling except as specifically set forth herein to be provided by Vendor.

Darüber hinaus treffen den Softwarebesteller Sorgfaltspflichten beim Umgang mit der Software²⁸¹:

Customer hereby agrees that at all times prior to making payment in full to Vendor, Customer shall:

- i) Keep the System free from all liens and encumbrances;*
- ii) Not use or permit the System, or any item, element or component thereof, to be used in any careless, reckless or negligent manner which is likely to be injurious to said products;*
- iii) Not make or permit any alterations to said products without Vendor's prior written consent; and*
- iv) Upon reasonable notice during regular business hours, permit inspection of the System by Vendor or Vendor's designated agent.*

Bei FP ergeben sich die besonderen Pflichten des Softwarebestellers aus der soeben dargestellten gemeinsamen Verantwortung bei der Projektleitung und der Mitwirkung bei der Spezifikation. In dem bestellerfreundlichen Vertrag AD sind keine besonderen Pflichten des Bestellers vorgesehen.

²⁸⁰ Im Purchase Agreement unter 9.

²⁸¹ Im Purchase Agreement unter 19 b); Unter 13. wird *software* als *system* definiert. Diese etwas unübersichtliche Vorgehensweise macht es verständlich, warum in einigen Verträgen (wie z.B. AE oder auch aus Gesetzen, wie dem U.C.C. bekannt) zu Beginn des Vertrages ein eigener Abschnitt den Definitionen gewidmet wird.

Das Element der Zusammenarbeit in FP wird durch eine Regelung bezüglich der Notwendigkeit von Berichten und gemeinsamer Besprechungen dokumentiert, die mindestens einmal im Vierteljahr stattfinden sollen²⁸²:

Progress Reports and Meetings for SMS and MAS. FDRI and PRODIGY agree to meet regularly as described in Exhibit 1 Section 3 to review progress of the MAS and SMS Systems Development and Maintenance. Such meetings shall be held as needed, but in any event not less frequently than quarterly and shall be documented in writing and signed by the Project Managers, and reflect all questions discussed and decisions made.

VI. Veränderungen (Change Management)

Das Change Management wird bei FP in zwei Regelungen behandelt. Selbst alltägliche Änderungen bedürfen der Zustimmung durch die Projektleitung, und substantielle Änderungen sollen nur in der Form schriftlicher Vertragsergänzungen erfolgen²⁸³:

Change orders. Day-to-day operational changes in the processing services specified in Exhibit 15 and otherwise provided for under this Agreement (e.g., print suppression; operational schedule changes) may only be made with approval of the Project Managers. Substantial changes in the processing services provided under this Agreement (e.g., eliminating the microfiche provided for in Exhibit 17; processing changes requiring coding) may only be made by written amendment hereto.

Für die Durchführung von Änderungen ist kein besonderes Verfahren vorgesehen. Es wird vielmehr einleitend die Verteilung der Prioritäten im Falle von Änderungen dem Softwarebesteller zugeordnet und die Notwendigkeit begründet, Änderungen nur nach vorheriger Schätzung der Auswirkung auf Kosten und Zeitplan durch das Softwarehaus einzuleiten. Der Tendenz zur gemeinschaftlichen Projektabwicklung folgend wird über Änderungen gemeinsam entschieden, und diese werden auf Basis eines gemeinsam zu entwerfenden Projektplans durchgeführt²⁸⁴.

Change Control. All proposed changes to SMS and MAS shall be prioritized by PRODIGY; provided, however, that all changes to SMS and/or to the MAS Design Baseline shall be mutually agreed upon and no changes shall be built into either SMS or MAS without prior estimation by FDRI of any cost

²⁸² Unter 1.1.6.

²⁸³ Unter 2.1.6 Change orders, als Unterpunkt von 2.1. Processing Services.

²⁸⁴ Unter 1.1.3.

and/or time frame implementation impact on the respective Joint Detailed Project Plans for SMS and/or MAS; and provided, further, that wherever costs of development or time-frames are at issue, the parties shall negotiate in good faith to reconcile the differences between them to reach, as soon as practicable, a mutually agreeable project plan to accommodate the implementation of any change. Change Control shall be administered for SMS and the MAS Design Baseline according to the methodologies and standards resulting from Exhibit 3 Section 10.

Bei AE ist im Falle von Änderungen ein einfaches, einstufiges Verfahren vorgesehen. Sie setzen ebenfalls eine schriftliche Änderungsvereinbarung voraus. Die Bestimmung geht dabei davon aus, dass der Softwarebesteller jederzeit eine Änderung verlangen und eine entsprechende Order geben kann, diese das Softwarehaus jedoch nur bei Einhaltung des Schriftformerfordernisses zu Vergütungsansprüchen berechtigt. Mögliche Kosten- oder Zeitplanauswirkungen werden nicht erwähnt²⁸⁵.

Company may at any time during the progress of the Work require additions to or alterations of or deductions or deviations (all hereinafter referred to as a "Change") from the Work called for under an Authorization Letter, subject to the terms of the applicable Authorization Letter. Except as provided for in the Authorization Letter, no Change shall be considered as additions to or deductions from the Work called for under the Authorization Letter, nor shall Supplier be entitled to any compensation for work done pursuant to, or in contemplation of, a Change, unless made pursuant to a written Change Order issued by Company and executed by both parties. Supplier shall be paid for any work performed in preparing a Change at Company's request.

OD enthält keine gesonderte Änderungsregelung²⁸⁶.

VII. Zusammenhang zwischen Leistungsbestimmung und Einstandspflicht

Wie bei den Vertragsmustern zeigen sich hinsichtlich der Einstandspflicht als Maß für die geforderte Leistung auch bei den ausgehandelten Verträgen überraschende Formulierungen.

Die flexibelste Regelung findet sich bei FP. Danach wird eine substantielle Übereinstimmung mit den vereinbarten Spezifikationen gefordert²⁸⁷:

²⁸⁵ Unter *Changes* in Section II.

²⁸⁶ Auf mögliche Änderungen abgezielt ist allenfalls die bereits genannte Regelung unter 5 d), wonach Unzulänglichkeiten bei der Spezifikation grundsätzlich nicht zu Lasten des Softwarebestellers gehen.

²⁸⁷ Unter 6.2 *Performance*, als Unterpunkt zu 6. *Warranties and Representation*.

Performance. FDRI represents and warrants that, as developed and accepted under this Agreement, the SMS System Enhancements and Maintenance changes and the MAS System shall substantially perform in accordance with all applicable Exhibits and shall be free of any defects which substantially affect the performance of the SMS System, as developed under the Prior Agreement. FDRI represents and warrants that those MAS System enhancements developed by FDRI in furtherance of this Agreement shall be free of any defects which substantially affect the performance of the SMS System, as updated by the SMS System Enhancements.

Bei AE bezieht sich die Einstandspflicht darauf, dass keine Kopiermechanismen vorgesehen sind und keine Virus- oder ähnliche Gefahr von der Software ausgeht. Darüber hinaus wird für die, ohne weiteres als notwendig vorausgesetzte, Behebung von Defekten auf *warranties* gemäß den *Authorization letters* verwiesen²⁸⁸:

Supplier shall provide such warranties for the repair of errors or defects in the Work as may be provided in the Authorization Letter applicable to such Work.

Supplier warrants that there are no copy protection or similar mechanisms within the Software which, either now or in the future, will interfere with the grants made in this Agreement. Supplier also warrants that Supplier shall not install, either now or in the future, any malicious code, program, or other internal component (e.g. software virus, software worm, software time bomb or similar component), which is designed to damage, destroy, or alter software, firmware, or hardware or which is designed, in any manner, reveal, damage, destroy, or alter any data or other information accessed through or processed by the Software in any manner or which is designed to adversely affect the operation of a computer or its memory. Supplier shall immediately advise Company, in writing, upon reasonable suspicion or actual knowledge that the Software provided under this Agreement may result in the harm described above. Supplier shall indemnify and hold Company and its customers harmless from any damage resulting from Supplier's failure to comply with this paragraph.

Die weitestgehende Bestimmung findet sich bei OD. Im Hauptvertrag wird im wesentlichen eine Abgrenzung von Drittprodukten und der nach dem Vertrag herzustellenden Software vorgenommen und die diesbezügliche Einstandspflicht ausgeschlossen²⁸⁹.

²⁸⁸ Unter *Warranty* in Section II, im Anschluss an das Zitat findet sich der übliche Ausschluss aller sonstigen Einstandspflichten: EXCEPT AS EXPRESSLY SET FORTH IN THIS SUBSECTION TITLED "WARRANTY", SUPPLIER DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION THE IMPLIED WARRANTY OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

²⁸⁹ Im Purchase Agreement unter 14. *Warranties*.

- a) *In addition to any express warranties set forth herein, Customer may be entitled to the benefit from certain limited warranties provided directly by the manufacturers, owners, publishers or distributors of these products. Vendor assumes no liability for, nor responsibility under, these third party warranties unless such liability or responsibility shall be specifically set forth herein.*
- b) *Customer hereby acknowledges and agrees that any warranties included herein do not extend or apply to Customer's use of any attachment, feature or software on or in conjunction with the System, or any item, element or component thereof, which has not been furnished pursuant to this Agreement or which has not been approved in writing by Vendor.*
- c) *Customer understands and agrees that Vendor, Vendor's supplier or suppliers, and the manufacturers, developers or distributors of the items, elements and components of the System are not engaged in a joint venture and, subject to Article 15, that Vendor has no intention, obligation or duty to warrant and/or represent the quality, performance and condition of such deliverables on their behalf.*

Im Anhang findet sich dann ein Anforderungsmaßstab, der kaum extensiver formuliert zu werden vermag²⁹⁰. Die Software soll danach u.a. nicht nur in Übereinstimmung mit den vereinbarten Anforderungen funktionieren sowie keine bekannten Fehler oder einen Virus enthalten, sondern alle gegenwärtigen und vernünftigerweise zu erwartenden Wünsche der Softwarebestellers erfüllen:

Vendor represents and warrants to Customer that: (a) the System will be capable of fulfilling all of Customers current and reasonably anticipated needs in an efficient and commercially desirable and competitive manner; (b) there are no problems or defects, including viruses, known to Vendor in the System; (c) the System does not contain any limitations which would render the System unsuitable for use by Customer in processing its own applications and those of its current divisions, subsidiaries or affiliates, or presently contemplated future divisions, subsidiaries or affiliates; (d) the System will operate and conform in all respects during the term hereof in accordance with the Performance Standards; (e) the System will meet all applicable local, state and federal codes, laws, regulations and tariffs, and that Vendor has no knowledge of any pending or proposed change in such codes, laws, regulations or tariffs which the System as installed would not meet.

²⁹⁰ *Rider Nr. 4*, ob dies ein uneingeschränktes Vertrauen des Softwarehauses in seine Produkte oder eine erhebliche Risikobereitschaft widerspiegelt oder das Ergebnis einer komplexen Verhandlung darstellt, ist nur für die unmittelbar am Vertragsschluss Beteiligten nachzuvollziehen.

Was die Preisgestaltung anbelangt, so findet sich bei OD eine Fixpreisvereinbarung²⁹¹, während bei FP monatliche Zahlung vorgesehen ist²⁹². Bei AE wird ebenfalls eine am Zeitaufwand orientierte Vergütung vereinbart²⁹³.

C. Zusammenfassung und Ergebnis

Die Verträge sind von sehr unterschiedlichen Ausgangssituationen geprägt. Dies gilt zum einen hinsichtlich der Bestimmbarkeit der zu erbringenden Leistung (Ergebnisvarianz), zum anderen bezüglich der ökonomischen Machtverhältnisse. Während bei OD das angestrebte Ergebnis als klar voraussehbar beurteilt wird, geht FP von einer Präzisierung im Projektverlauf aus. Bei AE kommt die starke wirtschaftliche Stellung des Softwarebestellers insofern zum Tragen, als dem Softwarehaus die Rolle eines „Zuarbeiters“ zugewiesen wird, der vom Besteller definierte Funktionen erfüllt.

Diese tendenzielle Ausrichtung spiegelt sich auch in den Mustern wieder. Die an den Interessen des Softwarehauses orientierten Muster unterscheiden sich in den entscheidenden Punkten grundlegend von denen, die sich an den Interessen des Softwarebestellers ausrichten²⁹⁴. So verlangen die letztgenannten Muster eine weitgehende Übernahme von Beratungspflichten, die von der Bedarfsanalyse bis hin zur Bedarfsermittlung durch das Softwarehaus reicht²⁹⁵. Auch die Einflussnahme auf Personalentscheidungen ist hier sehr weitgehend ausgestaltet. Umgekehrt verzichten die am Interesse des Softwarehauses orientierten Muster etwa vollständig auf ein Change Management und verlangen vom Softwarebesteller Prüfungen der erbrachten Leistungen, die ein erhebliches Know-How voraussetzen.

²⁹¹ Purchase Agreement Nr. 3.

²⁹² Unter 2.2.1.

²⁹³ Section II, *Audit*.

²⁹⁴ Diese - etwa aus der Differenz von Einkaufs- und Verkaufs AGB - bekannte Differenzierung ist aus Sicht des parteiischen Rechtsberaters eines der Beteiligten (selbst-) verständlich, kann jedoch zu erheblichen Friktionen bei der Vertragsverhandlung führen, da sie nicht von sachlichen Kriterien, sondern ausschließlich von der Machtposition der Parteien herrührt. Ob zudem durch Verhandlungsmacht durchgesetzte Vertragsbedingungen, bei der ohnehin nicht unerheblichen Gefahr des Scheiterns eines Softwareprojektes, geeignet sind, eine tragfähige Grundlage für eine Zusammenarbeit der Parteien zu bilden, darf mit guten Gründen bezweifelt werden.

²⁹⁵ Überspitzt formuliert: Das Softwarehaus muss herausfinden, was der Softwarebesteller braucht.

Die Muster gehen durchgehend von einem mehrstufigen Verfahren zur Erarbeitung der Spezifikationen aus. Der erste Schritt besteht in einer Ermittlung der Anforderungen. Die Durchführung des zweiten Schrittes - der von diesen Anforderungen zur technischen Spezifikation führt - wird in fast allen Mustern dem Softwarehaus zugeordnet. Die der Implementation vorgelagerte Prüfung der Spezifikation wird in vier von sieben Mustern als Aufgabe des Softwarebestellers betrachtet. Eine einheitliche Ausrichtung danach, welche Partei sinnvollerweise die eine oder andere Aufgabe übernehmen sollte, ist nicht erkennbar.

Der Offenheit des Ergebnisses (hohe Ergebnisvarianz) stehen an anderer Stelle der Praxisbeispiele (hier hinsichtlich des Projektteams, der Systemverfügbarkeit und der Berichtspflichten) genaue Vorgaben gegenüber, und es werden die Elemente der gemeinschaftlichen Leistungserbringung betont. Wo das Ergebnis klar bestimmbar scheint (geringe Ergebnisvarianz), liegt der Schwerpunkt auf der Kontrolle zeitgerechter Leistungserbringung. Auch die Regelung des Change Managements orientiert sich in den Praxisbeispielen an der Ergebnisvarianz. Wo sie gering ist, findet sich keine entsprechende Regelung. Ist sie dagegen hoch, schlägt sich dies in entsprechenden vertraglichen Bestimmungen nieder.

Die Muster enthalten, mit der benannten Ausnahme der an Softwarehausinteressen orientierten Beispiele, alle eine mehr oder weniger ausgefeilte, mehrstufige Regelung zum Change Management, wobei es Aufgabe des Softwarehauses ist, die Auswirkung von Änderungen auf Kosten und Zeitplan zu ermitteln.

Die Einstandspflicht ist bei den Praxisbeispielen von der Ergebnisvarianz abhängig. Bei hoher Ergebnisvarianz wird ein flexibles Kriterium gewählt, während bei geringer Ergebnisvarianz hohe Anforderungen an das Softwarehaus gestellt werden. Ist der Softwarebesteller in einer sowohl wirtschaftlich als auch informationstechnisch starken Position, so hat er die Ergebnisvarianz in der Hand und kann das Softwarehaus fast wie einen „Subunternehmer“ der eigenen Softwareabteilung einsetzen.

In den Mustern findet sich zur Einstandspflicht ein breites Spektrum von Varianten, die von „*acceptable implementation*“ und „*perform substantially*“ über „*comply with requirements*“ und „*conform in all material aspects*“ bis hin zu „*strict compliance*“ und schließlich „*system shall meet needs*“ reichen. Ein Zusammenhang mit der Ergebnisvarianz ist nicht erkennbar, wobei davon auszugehen ist, dass diese angesichts der überwiegend detaillierten Bestimmungen zum Spezifikationsverfahren und Change Management hoch eingestuft werden kann.

Bei hoher Ergebnisvarianz wird in den Praxisbeispielen zeitabhängige Vergütung gewählt, während bei geringer Ergebnisvarianz ein Fixpreis vorgesehen ist. In den Mustern spielt diese Differenzierung keine Rolle. Hier trifft man ausschließlich auf zeitunabhängige Vergütungsregeln (Fixpreise). Was die Zuordnung zu Vertragsarten anbelangt, so ist sie in den untersuchten Beispielen ohne Bedeutung. Es wird keine klare Einordnung erkennbar. Die Leistungsbeschreibung ist am Ergebnis ausgerichtet, wobei der Schwerpunkt der vom Softwarehaus geforderten Leistung unterschiedlich auf Service- und Lieferungselemente verteilt wird.

Alle Muster gehen davon aus, dass die Bereitstellung funktionsfähiger Software geschuldet ist und dies - hier mit Ausnahme der am Softwarebesteller orientierten Muster - durch die Erbringung von Serviceleistungen erreicht wird.

3. DEUTSCHER TEIL

Kapitel 5: Rechtliche Rahmenbedingungen in Deutschland

A. Einordnung und Bedeutung der Rechtsquellen

I. Rechtsprechung und Gesetz

Das Gewaltenteilungsprinzip geht historisch²⁹⁶ davon aus, dass ein lückenloses Gesetz vorliegt und der Richter als Subsumtionsautomat tätig wird²⁹⁷. Auch wenn sich dies als Illusion herausstellt, weil die notwendige Schematisierung zwangsläufig einen Verlust an Begriffsschärfe zur Folge hat²⁹⁸, so bleibt der Grundgedanke des deutschen Rechtes doch, dass der Jurist das Sozialgeschehen von Normen her aufzugliedern und zu verstehen sucht²⁹⁹. Die Rechtsfindung erfolgt durch Subsumtion nach Wortlaut, Auslegung oder Gewohnheitsrechtssätzen und erst, wenn sich hieraus nichts ergibt, nach richterlicher Rechtsschöpfung³⁰⁰. Bereits in der Anfang des 20. Jahrhunderts in der Wissenschaft geführten Diskussion wurden auf der Grundlage rechtsvergleichender Betrachtung (mit anglo-amerikanischem Recht) allerdings Stimmen laut, die eine freiere Judikatur forderten³⁰¹. Gemäß Art. 20 Abs. 3 GG ist die Gesetzgebung an die verfassungsmäßige Ordnung, die vollziehende Gewalt und die Rechtsprechung an Gesetz und

²⁹⁶ Schlosser, Grundzüge der Neueren Privatrechtsgeschichte, 9. Aufl., Heidelberg, 2001, S. 191; Coing, in: Staudinger, Einl zum BGB, 12. Aufl., Berlin, 1995, Rn. 202ff.

²⁹⁷ Sehr anschaulich wird dies durch die Modellkodifikation des Civil Law, dem Code Civil: Hier ist es dem Richter gem. Art. 5 ausdrücklich untersagt, anlässlich einer Entscheidung allgemeine und regelnde Ausführungen vorzunehmen. Das preußische ALR von 1794 bestimmt in seiner Einleitung in § 46: „*Bey Entscheidungen streitiger Rechtsfälle darf der Richter den Gesetzen keinen anderen Sinn beylegen, als welcher aus den Worten, und dem Zusammenhange derselben, in Beziehung auf den streitigen Gegenstand, oder aus dem nächsten unzweifelhaften Grunde des Gesetzes, deutlich erhellet.*“ Es ist jedoch davon auszugehen, dass der Gesetzgeber des BGB „*niemals an die Lückenlosigkeit der Kodifikation geglaubt*“ hat, Coing, in: Staudinger, Einl. zum BGB, Rn. 204. Auch sprechen einige Gründe dafür anzunehmen, die den Richtern des Kaiserreichs zugeschriebene Anhängerschaft zur Begriffsjurisprudenz sei durchaus weniger ausgeprägt gewesen als angenommen: Eckert, Jörn; AcP 2003, S. 863ff.

²⁹⁸ Dubischar, Roland; Grundbegriffe des Rechts; Stuttgart Berlin Köln Mainz 1968, S. 21.

²⁹⁹ Dubischar, a.a.O., S. 10 a. E./11.

³⁰⁰ Nach Stimmen in der Literatur – Staudinger / Coing, a.a.O., Rn. 206; Schlosser, a.a.O., S. 191 - besonders instruktiv ausgedrückt in Art. 1 des schweizerischen ZGB.

³⁰¹ Schlosser, a.a.O., S. 194 zur sog. „Freirechtsschule“, vgl. auch Staudinger/Coing, Rn. 205ff; Soergel/Hefermehl, Anh. § 133, Rn. 16 äußert die Befürchtung, eine zu starke Betonung der richterlichen Wertungsfreiheit könne dazu führen, „*dass die Rechtsprechung zu einer zusammenhanglosen Falljustiz abgleitet und zur Rechtsunsicherheit führt*“.

Recht gebunden. Die Bindung an Gesetz *und Recht* impliziert Gestaltungsspielräume bei Ausübung der rechtsprechenden Funktion, die von dem BVerfG deutlich bestätigt werden³⁰². Hier kommt auch zum Ausdruck, dass es eine Bindung an Präjudizien, d.h. an Entscheidungen anderer Gerichte, nicht gibt. Jedes Gericht kann - und muss - aufgrund der Sachlage im konkret zu entscheidenden Fall und auf Grundlage der geltenden Gesetze sein Urteil fällen. Die Rechtsprechung auch der oberen und obersten Gerichte gehört nicht zu den verbindlichen Rechtsquellen. Dies schließt eine Bindung etwa an ständige Rechtsprechung oder eigene Präjudizien unter dem Gesichtspunkt des Vertrauensschutzes gleichwohl nicht aus³⁰³. In der Praxis zeigt sich dann auch die verständliche Tendenz, Entscheidungen an dem prognostizierten Verhalten der oberen Gerichte auszurichten. Im Alltagsgeschäft des Richtens wird das Rad nicht jedes Mal neu *er-* sondern *gefunden*.

II. Das Prinzip Privatautonomie

Im deutschen Recht lassen sich die Kernbestandteile der Regelungen für Softwareentwicklungsverträge dem BGB entnehmen³⁰⁴. Anders als in den USA ist das Zivilrecht durch vom Bund ausgiebig genutzte Gesetzgebungskompetenz³⁰⁵ bundeseinheitlich geregelt. Das BGB, welches auch vor kurzem noch als Respekt gebietendes „Kulturdenkmal“ bezeichnet wurde³⁰⁶, ist vom Grundsatz der Privatautonomie geprägt. Vor dem Hintergrund unbehinderten Marktzuganges, der prinzipiell freien Verfügbarkeit von Waren und Dienstleistungen und der gleichberechtigten Marktstellung und Verhandlungsposition soll der Marktteilnehmer mit der Freiheit zur Gestaltung seiner rechtlichen

³⁰² Zuletzt im Nichtannahmebeschluss des BVerfG vom 4. August 2004, Az: 1 BvR 1557/01, Orientierungssätze 1b) und 1c) m.w.Nachw. Wenn Coing die Worte „Equal Justice under Law“ als gleichbedeutende Regel für deutsche und amerikanische Richter kennzeichnet (Staudinger / Coing, a.a.O., Rn. 213), so sollte freilich deutlicher werden, was in den Vereinigten Staaten gemeinhin unter „Law“ verstanden wird, nämlich das Recht einschließlich des Case Law als seiner bedeutendsten Quelle.

³⁰³ Staudinger / Coing, a.a.O., Rn. 229ff.

³⁰⁴ Nicht berücksichtigt werden hier die im Zusammenhang mit dem zentralen Thema der Urheberrechte stehenden Regeln aus dem UrhG sowie Einzelfragen aus dem Bereich des GWB.

³⁰⁵ Art. 74 Abs.1 Nr.1, 72 GG.

³⁰⁶ Flume, Vom Beruf unserer Zeit für Gesetzgebung, in: ZIP 2000, S. 1427 (1429).

Beziehungen ausgestattet werden³⁰⁷. Die Privatautonomie erlaubt es, das “Ob” (Abschlussfreiheit) und “Wie” (Inhaltsfreiheit) der Verträge zu bestimmen³⁰⁸. Soweit der Inhalt an Grenzen stößt, etwa durch die §§ 305ff BGB, sind Kaufleute – und um solche wird es sich bei den Vertragspartnern von Softwareentwicklungsverträgen ausnahmslos handeln – weniger schutzwürdig als Verbraucher³⁰⁹. Das zwingende Recht tritt gegenüber der autonomen Einschätzung von Chancen und Risiken durch den Unternehmer zurück³¹⁰. Gleichwohl stellt der Gesetzgeber mit dem Schuldrecht eine Regelungssystematik zur Verfügung, die sich bemüht, eine Grundlage und ein Auffangbecken für Verträge jeder Art zu bieten³¹¹. Es steht den Gestaltern eines Vertrages danach zwar die Möglichkeit offen, dessen Inhalt, nicht jedoch die Einordnung in die Systematik, verbindlich festzulegen³¹². Die rechtlichen Rahmenbedingungen für Softwareentwicklungsverträge hängen im deutschen Recht ausschlaggebend davon ab, in welche Vertragskategorie sie einzuordnen sind.

III. Auslegungsgrundsätze

Die Grundsätze für die Auslegung von Verträgen finden sich im deutschen Recht im BGB in den §§ 133 und 157 BGB. Diese ergänzen sich und sind

³⁰⁷ Ausführlich hierzu Schmidt, in: Esser, Josef/Schmidt, Eike, Schuldrecht Band I Allgemeiner Teil Teilband 1, 8. Aufl., Heidelberg, 1995, § 1 I S. 2ff (S.3).

³⁰⁸ Einschränkungen kommen hier v.a. unter dem Gesichtspunkt des § 20 GWB (kartellrechtliches Diskriminierungsverbot für marktbeherrschende Unternehmen), in Einzelfällen gestützt oder ersetzt durch § 826 BGB (Kontrahierungszwang bei rechtlicher oder tatsächlicher Monopolstellung) in Betracht, vgl. Hopt, in: Baumbach/Hopt, Handelsgesetzbuch, 31. Aufl., München, 2003, Rn. 6 f und 11 Einl. Vor § 343.

³⁰⁹ Die sich aus den §§ 134, 138 BGB ergebenden Grenzen gelten demgegenüber unabhängig von der Unternehmereigenschaft.

³¹⁰ Hopt, in: Baumbach/Hopt, a.a.O., Rn. 4 zu Einl. vor § 1; die tatsächliche Wirkung dieser Unterscheidung sollte freilich nicht überschätzt werden, selbst dort, wo Unternehmer ausdrücklich von der Anwendbarkeit bestimmter Regeln ausgenommen sind - wie z.B. nach § 310 Abs.1 S.1 BGB für die §§ 305 Abs. 2, 308, 309 BGB - findet die Inhaltskontrolle von Vereinbarungen im unternehmerischen Verkehr prinzipiell in gleicher Weise statt wie im Rechtsverkehr mit Verbrauchern, vgl. Erman/Roloff, Bürgerliches Gesetzbuch Handkommentar, 11. Aufl., Köln Münster, 2004, § 307 Rn. 35.

³¹¹ Ob dies heute noch gelingen kann, ist nicht einfach zu beantworten – so bemerkt etwa Weyers treffend: „Nun bietet bei aller Vielfalt und Farbigkeit früherer Gesellschaftsordnungen in anderen Bereichen der wirtschaftliche und technologische Rahmen, in dem diese Regeln für Kaufverträge entstanden, rezipiert und entwickelt worden waren, im Vergleich zu heute ein Bild eher idyllischer Einfachheit und Überschaubarkeit, und das gilt auch von dem objektiven Wirtschaftsmodell des BGB, also der Wirtschaftsstruktur, für die seine Regeln in vollem Umfang angebracht sein mochten.“ Weyers, in: Esser/Weyers, Schuldrecht Band II Besonderer Teil, 8. Aufl., Heidelberg, 1998, § 2 II 1 b) (S.6).

³¹² So wird ein Vertrag etwa nicht dadurch zum Werkvertrag, dass ich ihn als solchen bezeichne – vergleichbar einem Esel, der mit der Aufschrift “Kuh” versehen wird.

beide für die Vertragsauslegung anzuwenden³¹³, wobei § 133 BGB auf ein subjektives Element (den wirklichen Willen), § 157 BGB dagegen auf ein objektives Element (Treu und Glauben sowie die Verkehrssitte) zielt³¹⁴. Die Auslegung des vorhandenen, d.h. erklärten Parteiwillens folgt vorrangig den subjektiven Kriterien des § 133, während die ergänzende Vertragsauslegung nach § 157 sich anhand objektiver Kriterien auf einen hypothetischen Parteiwillen bezieht. Beide Vorschriften zusammen dienen dazu, den Inhalt der Leistungspflicht, das "rechtliche Wollen", zu ermitteln³¹⁵.

Für die Ermittlung der auslegungsrelevanten Umstände gelten in einem Prozess die Regeln über Behauptungs- und Beweislast³¹⁶. Die anschließende Ermittlung des Parteiwillens nach Auslegungsgrundsätzen stellt demgegenüber eine richterliche Würdigung dar, bei der keine Bindung an das Parteivorbringen besteht³¹⁷. Entscheidend ist bei der Auslegung das von den Vertragsparteien verfolgte Ziel der Vereinbarung. Es kommt auf die Interessenlage und den wirtschaftlichen Zweck des Geschäftes an³¹⁸. Auszugehen ist vom Wortlaut der Erklärung³¹⁹. Maßgeblich ist die Verständnismöglichkeit dessen, für den die Erklärung bestimmt ist³²⁰. Als für die Auslegung relevante Begleitumstände sind sodann das Gesamtverhalten der Parteien und v.a. die Entstehungsgeschichte (wie Vorverhandlungen, Abwicklung früherer Geschäfte³²¹, mündliche Äußerungen der Beteiligten bei den Vertragsverhandlungen³²², Angebotsunterlagen und

³¹³ Palandt/Heinrichs, Rn. 1 zu § 133; Jauernig/Jauernig § 133 Rn. 7; Wendtland in: Bamberger/Roth, Rn. 19 zu § 133, Rn. 2 zu § 157; AnwK/Looschelders, § 133 Rn. 2; Erman/Armbrüster, § 157 Rn. 2; Soergel/Hefermehl, § 133 Rn. 2; MüKo/Mayer-Maly/Busche, § 133 Rn. 19f.

³¹⁴ AnwK/Looschelders, § 133 Rn. 3.

³¹⁵ Erman/Armbrüster § 157 Rn. 3; wobei eine Trennung von dem "rechtlichen Sollen" nach § 242 praktisch schwer durchführbar ist und die Grenzen daher verschwimmen, Soergel/Hefermehl § 133 Rn. 3.

³¹⁶ Erman/Palm, § 134 Rn. 42; Soergel/Hefermehl, § 133 Rn. 35.

³¹⁷ Erman/Palm § 133 Rn. 41; Soergel/Hefermehl verweist dafür auf die Fragwürdigkeit des Indizwertes der Selbstinterpretation der Partei, § 133 Rn. 18 und 35.

³¹⁸ Erman/Palm, § 133 Rn. 32; Soergel/Hefermehl § 133 Rn. 25; ähnlich steht bei der Auslegung des Gesetzes die ratio legis im Vordergrund, Palandt/Heinrichs, Rn. 34, 38 Einl. vor § 1; Soergel/Hefermehl, Anh. § 133 Rn. 7; in anderen kontinentaleuropäischen Rechtsordnungen sind die Auslegungskriterien für Gesetze, anders als im deutschen Recht, ausdrücklich geregelt, vgl. etwa §§ 6,7 ABGB und Art. 12 Codice Civile.

³¹⁹ Palandt/Heinrichs, Rn. 14; Erman/Palm § 133 Rn. 23, Erman/Armbrüster, § 157 Rn. 5; Soergel/Hefermehl, § 133 Rn. 24.

³²⁰ Stellvertretend für diese sog. Lehre vom Empfängerhorizont: MüKo / Mayer-Maly/Busche, § 133 Rn. 10, die dabei allerdings dem inneren Willen des Erklärenden eine Bedeutung für die Auslegung zukommen lassen.

³²¹ Bamberger/Roth/Wendtland, § 133 Rn. 25; Palandt/Heinrichs § 133 Rn. 14ff.

³²² Soergel/Hefermehl § 133 Rn. 25.

Leistungsverzeichnisse³²³) heranzuziehen. Bei formbedürftigen Erklärungen sind dabei nach teilweise und v.a. in der Rspr.³²⁴ vertretener Auffassung nur diejenigen außerhalb der Urkunde liegenden Umstände verwertbar, die in der Urkunde zumindest angedeutet sind³²⁵.

Verträge müssen bei ihrer Auslegung nach § 157 darüber hinaus die Grundsätze von Treu und Glauben berücksichtigen, also im Hinblick auf ein Ergebnis gedeutet werden, welches „die berechtigten Belange beider Parteien angemessen berücksichtigt³²⁶ und mit den Anforderungen des redlichen Geschäftsverkehrs im Einklang steht“³²⁷. Beachtung verlangt nach § 157 weiter die zum Zeitpunkt des Vertragsschlusses³²⁸ geltende Verkehrssitte, die infolge ihres Charakters als Norm unabhängig von ihrer Kenntnis Gültigkeit beansprucht³²⁹. Unter Kaufleuten ist der Handelsbrauch in § 346 HGB ausdrücklich als Richtlinie für die Bedeutung und Wirkung von Handlungen aufgenommen.

Eine weitere Auslegungsregel findet sich in § 305c Abs. II BGB, wonach Zweifel bei der Auslegung zu Lasten des Verwenders von Allgemeinen Geschäftsbedingungen gehen. Der hier zugrunde liegende Schutzgedanke wird über Allgemeine Geschäftsbedingungen hinaus auf Geschäfte angewandt, bei dem der Vertragstext vom wirtschaftlich und intellektuell Überlegenen entworfen wurde³³⁰.

Die Auslegung muss stets den Grundsatz der Privatautonomie³³¹ beachten, darf also nicht zu einem Ergebnis führen, das den übereinstimmenden

³²³ MüKo/Mayer-Maly/Busche, § 133 Rn. 44.

³²⁴ Unter dem Stichwort der Andeutungstheorie: BGH 20. Dezember 1974 – V ZR 132/72 - BGHZ 63, 359; 23. März 1979 – V ZR 24/77 - BGHZ 74, 116; 25. März 1983 – V ZR 268/81 - BGHZ 81, 150, zuletzt 17. Februar 2000 – IX ZR 32/99; NJW 2000, 1569.

³²⁵ Palandt/Heinrichs § 133 Rn. 19, a.A. Bamberger/Roth/Wendtland, § 133 Rn. 26; Erman/Palm § 125 Rn. 16; MüKo/Mayer-Maly/Busche, § 133 Rn. 29, 53.

³²⁶ Ähnlich: Bamberger/Roth/Wendtland, § 157 Rn. 12.

³²⁷ Palandt/Heinrichs § 133, Rn. 20.

³²⁸ Bamberger/Roth/Wendtland, § 157 Rn. 20; Erman/Armbrüster, § 157 Rn. 30; MüKo/Mayer-Maly/Busche, § 133 Rn. 5.

³²⁹ MüKo/Mayer-Maly/Busche, § 157 Rn. 16.

³³⁰ Palandt/Heinrichs, § 133 Rn. 23, ebenso Bamberger/Roth/Wendtland, § 157 Rn. 13; a.A. Jauernig/Jauernig, § 133 Rn. 11; vgl. auch Bamberger/Roth/Schmidt, § 305c, Rn. 23, der auf den historischen Ursprung der Unklarheitenregel im römischen Recht hinweist: „*ambiguitas contra stipulatorem est*“; MüKo/Mayer-Maly/Busche, § 157 Rn. 7 bei „*Ausnutzung wirtschaftlicher Übermacht*“.

³³¹ Siehe oben S. 100.

Willen der Vertragsparteien in Frage stellt³³². Ein geäußelter Wille ist also stets zu respektieren und bei ergänzender Vertragsauslegung konsequent weiterzudenken³³³.

B. Anwendbares Recht

I. Einordnung des Vertragstypus

1. Notwendigkeit

Die Diskussion über das auf Softwareentwicklungsverträge anzuwendende Recht hat durch die Schuldrechtsreform neuen Auftrieb erhalten, der zu zahlreichen Stellungnahmen und auch umfangreicheren wissenschaftlichen Stellungnahmen Anlass gab. In vielen Beiträgen ist dabei die Notwendigkeit einer Einordnung des Vertragstypus gar nicht Bestandteil der Überlegungen. Hoeren betont in seiner Dissertation noch die Verbindlichkeit von Begriffskategorien, die durch den Gesetzgeber vorgegeben seien³³⁴. Die „wahre“ Rechtsnatur einer Vereinbarung, so wie die der Softwareüberlassung als Kauf³³⁵, stehe nicht zur Disposition der Vertragsparteien³³⁶. Von diesem Gedanken ist wohl auch Redeker ausgegangen, wenn er ausführt, Vertrag und Gesetz ließen nur wenig Regelungslücken³³⁷.

In einigen neueren³³⁸ und neuesten³³⁹ Stellungnahmen findet sich diese Auffassung nicht wieder. So betont Thewalt³⁴⁰ die Typenfreiheit des Schuldrechts und verweist auf die für Softwareentwicklungsverträge unbedeutenden zwingenden Vorschriften denkbarer Vertragstypen. Die

³³² Bamberger/Roth/Wendtland, § 157 Rn. 26, AnwK/Looschelders, § 157 Rn. 10.

³³³ Erman/Palm, § 133 Rn. 20.

³³⁴ Hoeren, Thomas, Softwareüberlassung als Sachkauf, Diss., Münster, 1989, S. 25f.

³³⁵ Hoeren, a.a.O., S. 30ff, 57.

³³⁶ Hoeren, a.a.O., S. 26.

³³⁷ Redeker, IT-Recht in der Praxis, 3. Aufl., München, 2003, S. 491.

³³⁸ Malzer, Hans-Michael, Der Softwarevertrag, Diss., Köln, 1991, S.3; Slongo, Doris, Der Softwareerstellungsvertrag, Diss., Zürich, 1991, S. 21; Schmid, Joachim; Probleme der Rechtsnaturbestimmung moderner Vertragstypen am Beispiel des „Back-Up-Vertrages“, CR 1994, S. 513; Volle, Peter; Rechtliche Einordnung der EDV-Systemverträge, CR 1996, S. 139.

³³⁹ Schubert, Michael von/Haase, Marina, Konfliktmanagement in IT-Projekten, ZKM 2003, S. 251; Thewalt, Stephan, Der Softwareerstellungsvertrag nach der Schuldrechtsreform, Diss., München, 2004, S. 11; Müller-Hengstenberg, CR 2005 S. 385; Seffer/Horter, ITRB 2005, S. 169.

³⁴⁰ Thewalt, a.a.O., 2004, S. 11.

unzureichenden gesetzlichen Vorgaben könnten nur eine pauschale Orientierungsfunktion³⁴¹ wahrnehmen, zumal das Projektleben nicht dem Gesetz folge³⁴².

Wenn Hoeren noch betont, das Recht solle sich nicht an der Technik orientieren, sondern „auf die Stärke seiner eigenen Denkweise vertrauen“³⁴³ und EDV-Fragen mittels des dem Zivilrecht „eigenen ´klassischen Instrumentariums´ lösen“³⁴⁴ so formulieren Malzer³⁴⁵ und Slongo³⁴⁶ bereits wenig später, dass Software eigene Regeln brauche. Dies unterstreicht Seffer, wenn er anmerkt, kein gesetzliches Leitbild passe³⁴⁷.

Im allgemeinen Vertragsrecht wird teilweise ausdrücklich von einem Ausschluss des dispositiven Rechts und der Notwendigkeit ergänzender Vertragsauslegung ausgegangen, wenn der gewählte Vertragstyp gesetzlich nicht geregelt ist oder der Interessenlage der Parteien nicht entspricht³⁴⁸.

2. Typenauswahl und Einordnungskriterien

Folgt man der in weiten Teilen der Fachliteratur betriebenen Suche nach einem gesetzlichen Typus, so lässt sich keine einheitliche Tendenz feststellen. In Betracht gezogen werden der reine Werkvertrag nach §§ 631ff BGB³⁴⁹ sowie die nach § 651 S.3 modifizierten, über § 651 S.1 BGB anzuwendenden Kaufvertragsregeln³⁵⁰. Vereinzelt finden sich auch eine vom

³⁴¹ Müller-Hengstenberg, CR 2005, S. 385.

³⁴² Sefferi/Horter, ITRB 2005, S. 169.

³⁴³ Hoeren, a.a.O., S. 57.

³⁴⁴ Hoeren, a.a.O., S. 171.

³⁴⁵ a.a.O., S. 3.

³⁴⁶ a.a.O., S. 21.

³⁴⁷ Seffer/Horter, ITRB 2005, S. 169.

³⁴⁸ Bamberger/Roth/Wendtland, § 157 Rn. 39; Erman/Armbrüster, § 157 Rn. 19; AnwK/Looschelders, § 157 Rn. 10ff, der in diesem Zusammenhang auch auf die Unbeachtlichkeit des dispositiven Rechts in den Fällen hinweist, in denen das in Frage stehende Gesetzesrecht veraltet ist, weil es den geänderten wirtschaftlichen Verhältnissen nicht mehr gerecht wird; MüKo/Mayer-Maly/Busche, § 157 Rn. 36, in Rn. 11 bemerken sie allerdings, dass Rechtsnormen „überhaupt der beste Zugang zur Erkenntnis dessen, was Treu und Glauben entspricht“ seien.

³⁴⁹ Malzer, a.a.O., S. 266; Thewalt, a.a.O., S. 66ff; Redeker, IT-Recht in der Praxis, S. 297; Herr, Robert, Anm. zu LG Karlsruhe, U.v. 13.05.1991, 10 O 458/89, CR 1992, S. 342; Müller-Hengstenberg, Claus Dieter, Vertragstypologie der Computersoftwareverträge, CR 2004, S. 161; Diedrich, Kay, Typisierung von Softwareverträgen nach der Schuldrechtsreform, CR 2002, S. 473.

³⁵⁰ Krauß, Wolfgang, Die Auswirkungen der Schuldrechtsreform auf Softwareverträge, Diss., Münster, 2004, S. 42, der den Grundcharakter des Werkvertrages betont; Marly, Jochen, Softwareüberlassungsverträge, 4. Aufl., München, 2004, S. 58; Schweinoch, Martin/Roas, Rudolf, Paradigmenwechsel für Projekte: Vertragstypologie der Neuerstellung von Individualsoftware, CR 2004, S. 326; Kottorf, Jost, Softwareerstellungsverträge nach der Schuldrechtsreform, K&R 2002, S. 105; Plath, Kai-Uwe, Abnahme bei Individualsoftwareverträgen, ITRB 2002, S. 98; Koch, Frank, Kaufrechtliche Vorgaben für

besonderen Schuldrecht gelöste Lizenzvertragslösung³⁵¹ sowie Hinweise auf das Gesellschaftsrecht³⁵² und einen sog. „dynamischen“³⁵³ bzw. evolutionären³⁵⁴ Vertrag

Die Kriterien, nach denen eine Zuordnung erfolgen soll, sind außerordentlich vielseitig. Zu den wenigen, soweit ersichtlich, einheitlich vertretenen Einordnungsgesichtspunkten zählt die Dauer der Softwareüberlassung³⁵⁵. Soll sie nur auf Zeit erfolgen, sei von Miete auszugehen. Für Softwareentwicklungsverträge ist eine Überlassung auf Zeit allerdings kaum denkbar³⁵⁶. Als ungeeignete Kriterien werden die Bezeichnung³⁵⁷ sowie die Vergütungsregelung (Einmalzahlung oder Zahlung nach Zeitabschnitten)³⁵⁸ aussortiert. Einige Stimmen wollen eine Abgrenzung danach vornehmen, wer das Herstellungsrisiko trägt³⁵⁹. Mehrfach vertreten ist das Einordnungskriterium der Angemessenheit der Normen³⁶⁰. Daneben werden die Bestimmtheit³⁶¹ und die Beeinflussbarkeit des Erfolges³⁶² genannt. Zwei

Verträge zur Softwareerstellung, ITRB 2002, S. 297; differenzierend danach, ob ein Anwender oder ein Softwarehaus als Besteller auftritt und wer die Verwertungsrechte erhält, wobei nur dann § 651 BGB greifen soll, wenn eine Anwender, der ein einfaches Nutzungsrecht erhält, Besteller ist: Redeker, Helmut, Softwareerstellung und § 651 BGB, CR 2003, S. 88; Schneider, Handbuch des EDV-Rechts, 3. Aufl., Köln, 2003, S. 1065 und 1137.

³⁵¹ Zuletzt Hilty, Reto, Der Softwarevertrag – ein Blick in die Zukunft, MMR 2003, S. 3 m.w.Nachw.

³⁵² Glöckler, in: Sobola, Sabine/Dobmeier, Gerhard (Hrsg.), Software- und Arbeitsverträge für die IT-Branche, Berlin, 2003, S. 230; Karger, „Kooperation“ bei komplexer Softwareentwicklung, ITRB 2003, S. 208.

³⁵³ Müller-Hengstenberg, CR 2005, S. 385.

³⁵⁴ Hoppner, Softwareentwicklungsverträge, Diss., Köln, 1997, S. 192ff.

³⁵⁵ Braun, Bernd, Überlassung von Computerprogrammen an den Anwender als Werkvertrag, BB 1992, S. 154; Marly, a.a.O., der seine Monographie danach gliedert.

³⁵⁶ Auch Marly erkennt diese Variante als „in der Praxis wenig verbreiteten Fall“, a.a.O., Rn. 44 (S. 21 Fn. 105).

³⁵⁷ Hoeren, a.a.O., S. 28 m.w.Nachw.

³⁵⁸ Schneider, Handbuch des EDV-Rechts, S. 1474.

³⁵⁹ Malzer, a.a.O., S. 289, der daneben darauf abstellt, dass bei der Softwareentwicklung eine Wertschöpfung für den Auftraggeber stattfindet; Schneider, Handbuch des EDV-Rechts, S. 1467, der weiter danach differenziert, wer bei der Projektsteuerung die Verantwortung, S. 1470ff bzw. insgesamt die Projektverantwortung trägt, Schneider, CR 2003, S. 317; Karger, Michael, „Kooperation“ bei komplexer Softwareentwicklung, ITRB 2004, S. 208; Diedrich spricht davon, dass mangels prüfbarer Leistungsgegenstände „Regelungen über die Risikoverteilung auf dem Weg zur Leistungserbringung“ erforderlich werden, die den Werkvertrag in den Vordergrund rücken, CR 2002, S. 473; ähnlich formuliert Kottorf, dass Softwareentwicklungsverträge für eine „angemessene Allokation der Projektrisiken“ zu sorgen hätten, was eindeutig zu einem Werklieferungsvertrag führe, K&R 2002, S. 105.

³⁶⁰ Köhler, Rechtsfragen zum Softwarevertrag, CR 1987, S. 827; Diedrich, CR 2002, S. 473; Müller-Hengstenberg, CR 2004, S. 161; Schweinoch/Roas, CR 2004, S. 326.

³⁶¹ Sobola/Dobmeier/Glöckler, a.a.O., S. 231.

³⁶² Thewalt, a.a.O., S. 114ff.

Stellungnahmen knüpfen für die Vertragseinordnung an die durch die Software zu verändernde Hardware an³⁶³.

Der BGH geht in ständiger Rechtsprechung von der Einordnung der Softwareentwicklung als Werkvertrag aus³⁶⁴, ohne dies ausführlich zu begründen. Die Einordnung habe den Besonderheiten der Vertragsgestaltung zu folgen, wobei darauf abzustellen sei, ob die Softwareüberlassung „im Rahmen eines Dauerschuldverhältnisses oder im Wege eines einmaligen Erwerbsaktes gegen einmaliges Entgelt erfolgt“³⁶⁵. Dieses Ergebnis wurde zementiert, nachdem der BGH die Sacheigenschaft von Software, wenn auch unter nur eingeschränkt verwertbarer Bezugnahme auf frühere Entscheidungen, eindeutig bejaht hatte³⁶⁶. Zur Einordnung zweifelhafter Verträge äußerte sich der BGH in jüngster Zeit am ausführlichsten außerhalb des Softwarebereiches. Stehe die Übertragung von Besitz und Eigentum im Vordergrund und bilde eine Montage nicht den Schwerpunkt des Vertrages, liege kein Werkvertrag vor³⁶⁷. Auch die Prognostizierbarkeit des Risikos³⁶⁸ und die Verantwortbarkeit des Erfolges³⁶⁹ wurden als Einordnungskriterium herangezogen. In der letztgenannten Entscheidung hält das Gericht zudem für erwähnenswert, dass es den Interessen beider Parteien entspreche, wenn die Nachbesserung durch den „mit den Verhältnissen am besten vertrauten Auftragnehmer“ geschähe, womit Werkvertragsrecht sach- und interessengerecht wäre³⁷⁰. Auch die übrige Rechtsprechung wendet ganz überwiegend, wenn auch meist ohne Begründung, Werkvertragsrecht an³⁷¹. Das Werkvertragsrecht passe besser, weil die Verjährung an die Abnahme und nicht an die

³⁶³ Koch, ITRB 2002, S. 297; Braun, BB 1992, S. 154, der von der „Konditionierung“ der Hardware spricht.

³⁶⁴ WM 1971, S. 615; NJW 1987, S. 1259; NJW 1990, S. 3011; NJW 1996, S. 1745; NJW-RR 1998, S. 1128; NJW-RR, S. 347; NJW 2001, S. 1718; CR 2002, S. 93.

³⁶⁵ U.v. 04. November 1987 - VIII ZR 314/86, Juris KORE305838701 Rn. 16.

³⁶⁶ U.v. 14. Juli 1993 – VIII ZR 147/92, Juris KORE372139300 Rn. 23.

³⁶⁷ U.v. 22. Juli 1998 – VIII ZR 220/97, Juris KORE306249800 Rn. 16 – Förderanlage; ebenso U.v. 15. April 2004, VII ZR 291/03, Juris KORE311192004 Rn. 11 – Mobilheim.

³⁶⁸ U.v. 16. Juli 2002 – X ZR 27/01, CR 2003, S. 244 (245) – F&E-Vertrag, wo ein Werkvertrag allerdings auch bei hohem Erfolgsrisiko - infolge der Vertragshoheit der Parteien - nicht ausgeschlossen wird.

³⁶⁹ U.v. 7. März 2002 – III ZR 12/01, NJW 2002, S. 1571 (1573) – Buchhaltung – .

³⁷⁰ BGH, a.a.O. NJW 2002, S. 1572.

³⁷¹ OLG Oldenburg U.v. 14.8.87, CR 1988, S. 214; LG Frankfurt, B.v.15.12.88, ECR LG S. 79; LG Karlsruhe, B.v. 13.5.91, CR 1991, S. 544; OLG Koblenz, U.v. 4.10.91, NJW-RR 1992, S. 688; OLG Köln, U.v. 26.6.92, OLGR Köln 1993, S. 237; und U.v. 25.6.93, OLGR Köln 1993, S. 270; OLG München, U.v. 31.1.95, OLGR München 1995, S. 217; OLG Düsseldorf, U.v. 18.7.97, CR 1997, S. 732; OLG Frankfurt, U.v. 3.5.01, OLGR Frankfurt 2001 S. 307; OLG Karlsruhe, U.v. 16.8.02, CR 2003, S. 95.

Ablieferung anknüpfe³⁷². Wenn eine Lösung geschuldet sei und der Schwerpunkt der vertraglichen Vereinbarung in Erstellung von Individualsoftware liege, so könne von einem Werkvertrag mit dienstvertraglichen Elementen ausgegangen werden³⁷³. Software verbleibe nicht im Immateriellen, vielmehr führe der Inhalt der geistigen Leistung über den Gebrauch zu gegenständlichen und fassbaren Ergebnissen³⁷⁴. Diese materialisierte geistige Leistung bedinge eine Anwendbarkeit des Werkvertragsrechts³⁷⁵.

b) Was ist Software

Nach § 651 BGB finden auf Verträge, die die Lieferung herzustellender Sachen zum Gegenstand haben, die Vorschriften über den Kauf Anwendung. Handelt es sich um eine nicht vertretbare Sache, so wird das Kaufrecht durch fünf Normen des Werkvertragsrechtes modifiziert. Vertretbare Sachen sind nach § 91 BGB Sachen, die im Verkehr nach Zahl, Maß oder Gewicht bestimmt zu werden pflegen, worunter Ware aus Serienfertigung³⁷⁶ zu verstehen ist. Nicht vertretbare Sachen sind solche, die infolge von Bestellerwünschen individuelle Merkmale aufweisen und daher schwer oder gar nicht anderweitig absetzbar sind³⁷⁷.

Wird Software als Sache eingeordnet, findet nach der aktuellen Fassung des § 651 BGB Kaufvertragsrecht Anwendung. Die hier untersuchte Individualsoftware wird, soweit die Sacheigenschaft angenommen wurde, ausnahmslos als nicht vertretbar angesehen.

Ob Software eine Sache im Sinne des Gesetzes darstellt, wird in der Literatur bis in die jüngste Zeit³⁷⁸ kontrovers diskutiert.

Bejaht wird es vor allem mit dem Hinweis auf die Notwendigkeit der Verkörperung: Zur Verkörperung des Programms sei die Funktionsfähigkeit des Computers zwingend, und ohne Speicherung könne kein Programm

³⁷² OLG Hamm, a.a.O., CR 1992, S. 206.

³⁷³ OLG Karlsruhe, Juris KORE541772003, Rn. 42ff.

³⁷⁴ OLG Hamm, U.v. 8.3.89, NJW 1990, 1609-1610 in Form von Ausdrucken und Bildschirmauszügen.

³⁷⁵ LG Karlsruhe, U.v. 13.5.93, CR 1991, 544-547.

³⁷⁶ Palandt/Heinrichs; § 91 Rn. 2.

³⁷⁷ Palandt/Sprau, § 651 Rn. 8.

³⁷⁸ Vgl. die Dissertationen von Krauß einerseits (a.a.O., Software ist eine Sache – S.18) und Thewalt andererseits (a.a.O., Software ist keine Sache – S. 45ff, (48)).

seine vorgegebenen Funktionen ausführen. Dem Begriff Software sei eine Materialisierung immanent³⁷⁹. Was im Falle von Software veräußert werde, sei „weder ein immaterielles, gegenstandsloses geistiges Gut, noch ein Nutzungsrecht, sondern eine Ware, nämlich das verkörperte und dokumentierte Wirtschaftsgut Software“³⁸⁰.

Die Gegenmeinung geht davon aus, dass das Programm und seine Verkörperung voneinander getrennt werden müssten³⁸¹. Die Nutzung von Software geschehe, anders als etwa bei einem Buch, an „unterschiedlichen physischen Stellen“ und hänge nicht an einer „festen unveränderten physischen Plattform, sondern wandert“³⁸². Es werde kein „materialisierter Gegenstand übertragen“, sondern das geistige Produkt erworben, welches primär Informationsgehalt habe³⁸³.

c) Zweck der Softwareentwicklung

Der Zweck der Softwareentwicklung wird, je nach Einschätzung von Software, unterschiedlich bewertet. Während eine Meinung in erster Linie an den Gedanken der Problemlösung anknüpft³⁸⁴, stellt eine andere auf die

³⁷⁹ Krauß a.a.O., S. 16f, Software könne auch nicht geistig funktionieren, ihre Funktionsweise sei ein „*rein physikalisch beschreibbarer auf eine Maschine einwirkender Vorgang*“, dabei müssten Werkverkörperung und urheberrechtliches Werk allerdings strikt getrennt werden (S. 20); anders dazu Hoeren: Eine Trennung von Programm und Datenträger sei nicht möglich (a.a.O., S. 32), auch interessiere den Erwerber nicht alleine die Funktionalität des Programmes (S. 33); Marly, a.a.O., Rn. 101: Software müsse unabdingbar in einer verkörperten Form existieren, sie bedürfe anders als etwa Know-How einer Form, die unkörperliche Überlassung z.B. über das Internet sei lediglich ein „spezieller Vertriebstyp“ (Rn. 109), der letztlich auch eine Verkörperung erfordere (Rn. 110).

³⁸⁰ Malzer, a.a.O., S. 366, das Werk bei einem Softwareentwicklungsvertrag sei nicht die geistige Leistung, sondern das Produkt dieser Leistung, deren Ergebnis (S. 269), es stehe hinter jedem Werk, wenn auch untergeordnete, geistige Arbeit (S. 268).

³⁸¹ Thewalt, a.a.O., S. 46f: Programm und Träger verhielten sich wie die Prägung zur Münze: „*Das Programm ist der durch den Prägestempel hervorgerufenen Prägung gleichzusetzen. Die Münze entspricht dem Datenträger.*“ Diese Trennung wird freilich auch von einem Anhänger der Sacheigenschaft vertreten, siehe Krauß, a.a.O., Fn. 80; Redeker führt entsprechend aus, die Programme existierten unabhängig von der konkreten Verkörperung (IT-Recht, Rn. 280; vgl. auch Diedrich, CR 2002, S. 473, wonach alleine das Speichermedium wahrnehmbar sei).

³⁸² Müller-Hengstenberg, CR 2004, S. 161, was sich insbesondere bei im Netzwerk verbundenen Systemen oder dem Outsourcing (z.B. in Form externer Rechenzentren) zeige; nach dem Sachbegriff des BGB müssten im Raum abgrenzbare physische Körper vorliegen; ebenso Thewalt, a.a.O., S. 47.

³⁸³ Redeker, CR 2003, S.88.

³⁸⁴ „*Zweck des Vertrages ist die Lösung von Anwenderproblemen im Bereich der Technik, Wissenschaft und Wirtschaft*“, Slongo, a.a.O., S. 7; Müller-Hengstenberg spricht davon, das Hauptinteresse der Vertragsparteien läge in der „Funktionsfähigkeit der Lösung“, CR 2004, S. 161; Ähnlich komme es den Parteien nach Malzer, a.a.O., S. 266, auf die

Veränderung einer Sache³⁸⁵ ab. Das Vertragsziel läge in der „Funktionsfähigkeit einer bestimmten Hardware“³⁸⁶. Aus dieser Erfolgsbezogenheit wird von der Rspr. eine bedeutsame Konsequenz abgeleitet: Eine wirksame Abnahme setze eine gewisse Zeit fehlerfreier Arbeit der Software voraus³⁸⁷. Es sei zu Gunsten des Bestellers eine Testphase erforderlich³⁸⁸. Die Brauchbarkeit von Software sei, wie bei einem Lizenzvertrag über ein Fertigungsverfahren oder eine Erfindung, erst nach langen Ausführungsversuchen erkennbar³⁸⁹.

d) Urheberrechtliche Einflüsse

Die Bedeutung des urheberrechtlichen Elementes der Softwareüberlassung wird sehr unterschiedlich beurteilt. Während sie nach einer Auffassung ohne Bedeutung für die Typeneinordnung ist³⁹⁰, stellt sie nach anderer Auffassung das zentrale Kriterium dar³⁹¹. Nach vermittelnder Ansicht ändert sie zwar an der Hauptpflicht - des werkvertraglichen Erfolges - nichts, doch führe die Pflicht zur Rechteverschaffung zu einem Typenkombinationsvertrag mit lizenzvertraglichen Element hinsichtlich der urheberrechtlichen Aspekte³⁹².

„Fertigstellung der funktionstauglichen Software (...) und somit auf einen messbaren Erfolg an“; Schweinoch/Roas nennen als „Endziel die Überlassung funktionsfähiger Software“ (CR 2004 S. 326); „Softwarebasierte Problemlösung“ nach Schmidl, Michael, Softwareerstellung und § 651 BGB – ein Versöhnungsversuch, MMR 2004, S. 590.

³⁸⁵ Nach Koch, ITRB 2002, S. 297, ist Vertragszweck die „Veränderung des Rechners“. Der Computer werde „in seinen wesentlichen Eigenschaften verändert, nämlich in seinen Problemlösungsmerkmalen“.

³⁸⁶ Braun, BB 1992, S. 154; Hilty, der die Sacheigenschaft von Software aus der Perspektive eines Urheber- und Patentrechters vehement zurückweist, will „dynamischen“ (gegenstandslosen) und „statischen“ Gebrauch unterscheiden. Dynamischer Gebrauch sei die „Nutzung von Funktionalität, wie sie nur Erfindungen haben“, MMR 2003, S. 3.

³⁸⁷ OLG Köln, in: OLGR Köln 1992, S. 237.

³⁸⁸ OLG München, in: OLGR München, 1995, S. 217.

³⁸⁹ BGH, U.v. 04.11.87, BGHZ 102, S. 135; vor der Möglichkeit zur Erprobung sei keine Abnahme möglich, U.v. 02.11.95, CR 1996, S. 667, und es kann eine Umkehr der Beweislast stattfinden, wenn Mängel typischerweise erst spät erkennbar sind, U.v. 02.07.96, BGHZ 133, S. 155 allerdings für die Sondersituation sich als fehlerhaft erweisender Backups.

³⁹⁰ Schweinoch/Roas, CR 2004, S. 326: Nutzungsrechte sind nach dem Gesetzeswortlaut des § 651 BGB irrelevant, auch würde bei zugestandenem Einfluss die Typisierung wandelbar, da von - häufig vorkommender - Vertragsgestaltung abhängig.

³⁹¹ Hilty, MMR 2003, S. 3.

³⁹² Thewalt, a.a.O., S. 167.

3. Ergebnis

So eindeutig die Einordnung von Softwareentwicklungsverträgen in der Rspr. ausfällt, so uneinheitlich stellt sie sich in der Literatur dar. Nach der Schuldrechtsreform hat sich dieser Sachverhalt nicht geändert. Von der wohl überwiegenden Meinung wird modifiziertes Kaufvertragsrecht nach § 651 S.3 BGB angewandt, wenn auch einige Stimmen reines Werkvertragsrecht nach §§ 631ff BGB anwenden wollen. Einzelstimmen sprechen sich für eine lizenzvertragliche Lösung oder einen „dynamischen“ Vertrag aus. Einzelne Autoren schließlich betonen die Unmaßgeblichkeit der schuldrechtlichen Modelle für die Vereinbarungen der Parteien.

II. Der Mangelbegriff bei der Softwareentwicklung

Für die Leistungsbestimmung ist es von besonderer Bedeutung, welcher Maßstab bei der Softwareentwicklung angelegt werden muss, ob sie also wie jedes andere Produkt anzusehen oder eine besondere Betrachtungsweise gerechtfertigt ist.

Ein modifizierter Fehlerbegriff wird von manchen Stimmen in der Literatur abgelehnt. Auch bei anderen hochtechnisierten Industrieprodukten stelle sich das Problem angeblicher Unvermeidbarkeit von Fehlern, und für die Softwareindustrie sei vom Gesetzgeber keine Privilegierung vorgesehen³⁹³. Einige Stimmen der Literatur vertreten allerdings die Gegenauffassung. So führt bereits Slongo aus, Individualsoftware sei mit (nur) zufriedenstellender Gebrauchstauglichkeit abnahmebereit³⁹⁴, und mit Fehlern etwa wegen des Nichterfassens einer im Betrieb möglichen Aufgabenstellung müsse immer gerechnet werden³⁹⁵. Im Softwarebereich gäbe es keine mangelfreie Leistung, sondern „maximal eine annähernd fehlerfreie“³⁹⁶.

Diese Einschätzung wird von einigen Gerichten geteilt. So führt das LG Mannheim aus, bei komplizierter Technik wie Individualsoftware sei mit Problemen zu rechnen³⁹⁷, das OLG Oldenburg bestätigt, es sei

³⁹³ Krauß, a.a.O., S. 60f; ebenso bereits Bartl, Rechtliche Problematik der Softwareverträge, BB 1988, S. 2122: Es sei Aufgabe des Anbieters der Software, Leistungseinschränkungen zu offenbaren.

³⁹⁴ a.a.O., S. 44.

³⁹⁵ a.a.O., S. 51.

³⁹⁶ Sobola/Dobmeier, a.a.O., S. 33; ebenso deutlich Sick, Ulrich, Verträge im Projekt- und Systemgeschäft, 2. Aufl., Heidelberg, 2004, S. 48: Software kann nicht fehlerfrei sein.

³⁹⁷ Urteil v. 20.06.86, in: ECR LG, S. 1-2.

bekanntermaßen schwierig, Individualsoftware auf Anrieb fehlerfrei zu erstellen³⁹⁸, und das LG München stellt fest, nach tatsächlichen Gegebenheiten und üblichen Vertragsbedingungen garantiere kein Hersteller Fehlerfreiheit³⁹⁹.

III. Leistungsbestimmung

1. Bedeutung insbesondere bei der Softwarestellung

Die Bedeutung der Leistungsbestimmung wird in der Literatur mit großer Deutlichkeit unterstrichen⁴⁰⁰. Es sei insbesondere vor dem Hintergrund eines im Rahmen der Schuldrechtsreform erweiterten Fehlerbegriffes äußerst wichtig, den Leistungsumfang genau zu vereinbaren⁴⁰¹. Das Schwergewicht des Vertrages sollte bei der eigentlichen Leistung und deren Spezifikation liegen⁴⁰². Die Notwendigkeit genauer Zieldefinition im Hinblick auf die zu erbringenden Funktionen, den Aufwand und die Zeit sei auch in der IT-Branche vorhanden⁴⁰³. Teilweise wird das Vorliegen einer werkvertraglichen Einigung ausdrücklich an das Vorhandensein genauer Leistungsbestimmung gebunden⁴⁰⁴.

2. Zeitpunkt und Anforderungen

Als Zeitpunkt für die Erstellung der Leistungsbestimmung wird in der Literatur eine häufige Verspätung, d.h. eine Formulierung nach Vertragsschluss, diagnostiziert. Es sei üblich, Spezifikationen erst im Projektverlauf vorzunehmen⁴⁰⁵. Die Realisierung weiche zudem häufig von der anfänglich

³⁹⁸ Urteil v. 14.08.87, CR 1988, S. 214.

³⁹⁹ Urteil v. 18.11.88, NJW 1989, S. 2625; Nach dem OLG Nürnberg ist das Ergebnis häufig fehlerbelastet, U.v. 08.11.95, ECR OLG 213, S. 799.

⁴⁰⁰ Vgl. Thewalt, a.a.O., S. 128 m.w.Nachw.; Allgemein ist es bei einer individuellen Wertschöpfung erforderlich, eine „brauchbare und erschöpfende“ Leistungsbeschreibung zur Verfügung zu stellen, MüKo / Soergel, § 631 Rn. 132.

⁴⁰¹ Sick, a.a.O., S. 14: Dies sei eine der wichtigsten Aufgaben bei der Vertragsformulierung und -verhandlung (S. 15).

⁴⁰² Schneider, Handbuch, a.a.O., Vorwort S. VI.

⁴⁰³ Müller-Hengstenberg, BVB-EVB-IT-Computersoftware, 5. Aufl., Berlin, 2001, S. 178; der weiter ausführt, diese Notwendigkeit werde jedoch nicht immer angemessen beachtet, was zu „beklagenswerten Missständen“ führe.

⁴⁰⁴ In Form eines Pflichtenhefts, Malzer, a.a.O., S. 276f, zum Pflichtenheft siehe auch sogleich.

⁴⁰⁵ Schneider, Handbuch, a.a.O., S. 1485.

konzipierten Lösung aus vielen Gründen ab⁴⁰⁶. Selbst wenn ein Pflichtenheft erstellt werde, erweise es sich nicht selten als unzureichend⁴⁰⁷. Es sei „an der Tagesordnung“, dass die tatsächlichen Bedürfnisse nicht den definierten Anforderungen entsprechen⁴⁰⁸.

Der BGH weist darauf hin, dass die Bedeutung der Leistungsbeschreibung im Einzelfall ermittelt werden muss⁴⁰⁹. Zudem könne auch eine Leistung geschuldet sein, die nicht ausdrücklich vereinbart wurde⁴¹⁰.

3. Gesetzliche Regeln nach modifiziertem Kaufvertragsrecht (§ 651 Abs.1 S.3 BGB) und Werkvertragsrecht (§§ 631ff BGB)

Wird über § 651 Abs.1 S. 3 modifiziertes Kaufvertragsrecht angewandt, so besteht die Hauptpflicht des Softwareerstellers nach § 433 Abs.1 BGB in der Übergabe und Eigentumsverschaffung frei von Sach- und Rechtsmängeln. Das Werkvertragsrecht verpflichtet demgegenüber zur Herstellung des versprochenen Werkes (§ 631 Abs.1 BGB), die nach § 633 Abs. 1 BGB ebenfalls mangelfrei zu erfolgen hat. § 631 BGB erfasst ein Vertragsmodell mit der größten Vielfalt von Lebenssachverhalten⁴¹¹, wobei der Schwerpunkt seiner Anwendung auf dem Bausektor liegt⁴¹². Vertragsgegenstand ist jedenfalls eine entgeltliche Wertschöpfung⁴¹³. Diese kann nach § 631 Abs.2 BGB neben der Herstellung oder Veränderung einer Sache auch in einem anderen Erfolg bestehen⁴¹⁴. Im Vordergrund steht im Unterschied zum Kaufvertrag nicht die Übertragung von Eigentum und Besitz, sondern das Ergebnis einer Tätigkeit des Unternehmers⁴¹⁵.

⁴⁰⁶ Redeker, IT-Recht, a.a.O., Rn. 436.

⁴⁰⁷ Schaub, CR 1993, S. 329, vgl. auch Müller-Hengstenberg, CR 2005, S. 385.

⁴⁰⁸ Schubert/Haase, ZKM 2003, S. 251.

⁴⁰⁹ U.v. 16.7.2003, BGHZ 151, S. 330 bezüglich eines Forschungs- und Entwicklungsvertrages.

⁴¹⁰ U.v. 23.1.1996, NJW 1996, S. 1745.

⁴¹¹ Weyers, Typendifferenzierung im Werkvertragsrecht, AcP 182 (1982) S. 60.

⁴¹² MüKo / Soergel, § 631 Rn. 2.

⁴¹³ MüKo / Soergel, § 631 Rn. 5; diesen gängigen Begriff hält Voit für wenig hilfreich und verweist, allerdings mit wenig Konkretisierungsgewinn, auf eine typologische Abgrenzung nach Fallgruppen und die Angemessenheit der Rechtsfolgen, Bamberger/Roth / Voit, § 631, Rn.4.

⁴¹⁴ Dies zielt hauptsächlich auf unkörperliche Werke, „in denen das Werk lediglich als Abbildung in der Gedankenwelt existiert und es an jeder gegenständlichen Beziehung fehlt“ wie bei Theateraufführungen und Konzerten), AnwK / Raab, Bd. 2 Tb 2, § 631 Rn. 11.

⁴¹⁵ Raab nennt Kaufverträge reine Sachverschaffungsverträge, hier stünde die Beschaffenheit der Sache im Vordergrund, während die Fähigkeiten des Verkäufers unerheblich blieben. Beim Werkvertrag würde der Unternehmer dagegen gerade wegen

Unterschiede ergeben sich nach der Schuldrechtsreform nicht mehr hinsichtlich der Voraussetzungen und dem Umfang von Schadensersatz sowie im Grundsatz bei den Voraussetzungen für das Vorliegen eines Mangels und dessen Rechtsfolgen⁴¹⁶. Eine bedeutsame Ausnahme stellt dabei allerdings das Wahlrecht zwischen Nachbesserung und Neuerstellung dar. Im modifizierten Kaufvertragsrecht liegt es gem. § 439 Abs. 1 BGB beim Käufer, im Werkvertragsrecht beim Unternehmer⁴¹⁷. Darüber hinaus kann der Besteller beim Werkvertrag anders als bei modifiziertem Kaufrecht den Mangel selbst beseitigen und den Ersatz der erforderlichen Aufwendungen verlangen (§ 634 Nr. 2 BGB). Dies bezeichnet Schneider als die „wahrscheinlich gefährlichste Waffe“ des Bestellers von Software⁴¹⁸. Der bedeutsamste Unterschied ist die Notwendigkeit (gem. § 640 beim Werkvertrag) bzw. Entbehrlichkeit (modifizierter Kaufvertrag) der Abnahme. An sie sind ein Bündel zentraler Rechtspflichten geknüpft, nämlich die Fälligkeit der Vergütung (§ 641 BGB), der Gefahrübergang (§ 644 Abs. 1 BGB), der Beginn der Verjährung (§ 634a Abs. 2 BGB⁴¹⁹) sowie der Gewährleistungsausschluss bei Mangelkenntnis (§ 640 Abs. 3 BGB). Mit der Abnahme endet, vorbehaltlich etwaiger Nacherfüllungsansprüche, das Erfüllungsstadium⁴²⁰, und der Anspruch auf den Werklohn ist nunmehr unabhängig vom weiteren Schicksal des Werkes⁴²¹. Es findet eine scharfe

seiner Sachkunde ausgesucht, AnwK / Raab, Rn. 15.

⁴¹⁶ Eine Ausnahme ergibt sich aus § 434 Abs. 1 S. 2 und Abs. 2 BGB. Danach sind öffentliche Äußerungen des Verkäufers, Herstellers oder seiner Gehilfen aus der Werbung oder bei der Kennzeichnung für die Soll-Beschaffenheit grundsätzlich zu berücksichtigen. Auch kann eine unsachgemäße Montage oder mangelhafte Montageanleitung zu einem Mangel führen.

⁴¹⁷ Eine Entscheidung, die darin begründet liegt, dass der Unternehmer bei der Werkerstellung auf Leistungshindernisse mit der Folge größerer Unsicherheit auf seiner Seite stoßen kann, und zudem über die überlegene Sachkunde bei der sinnvollen Art der Nacherfüllung verfügt, Bamberger/Roth / Voit, § 631 Rn. 2.

⁴¹⁸ Schneider, Handbuch, a.a.O., S. 1114, gemildert werde das Risiko allerdings durch das Verweigerungsrecht des Unternehmers gem. § 635 Abs. 3 und 275 Abs. 2 BGB bei grobem Missverhältnis des Aufwandes zum Leistungsinteresse bzw. unverhältnismäßigen Kosten, S. 1117.

⁴¹⁹ Wobei dies (nur) für die Fälle gilt, in denen der geschuldete Erfolg in der Herstellung, Wartung oder Änderung einer Sache oder in der Erbringung von Planungsleistungen hierfür, einem Bauwerk oder entsprechender Planungs- oder Überwachungsleistungen besteht.

⁴²⁰ Vgl. AnwK / Raab, § 640 Rn. 2; die vertraglichen Beziehungen schlagen bei Akzeptanz des Werkes als im wesentlichen vertragsgemäße Leistung insoweit um, als dass nunmehr die Mängelbeseitigung und nicht mehr die Erstellung eines mangelfreien Werkes geschuldet ist, Bamberger/Roth / Voit, § 640 Rn. 1.

⁴²¹ Vgl. AnwK / Raab, § 644 Rd. 9.

zeitliche Trennung der Risikosphären statt⁴²². Die Verjährung ist zudem in § 634a BGB anders geregelt als in § 438 BGB für das Kaufrecht. Nach § 438 Abs.1 Nr. 3 BGB gilt für die hier in Frage stehenden Ansprüche eine Verjährungsfrist von zwei Jahren. Bei § 634 Abs. 1 Nr. 1 BGB verjähren Ansprüche ebenfalls in zwei Jahren, allerdings setzt dies nach dem Tatbestand voraus, dass der Erfolg in der Herstellung, Wartung oder Veränderung einer Sache oder in der Erbringung von Planungs- oder Überwachungsleistungen hierfür besteht. Geht man mit einem Teil der Literatur davon aus, Software sei keine Sache, so führt dies nach § 634a Abs.1 Nr. 3, 195 BGB zu einer Verjährung von drei Jahren, die gem. § 199 Abs.1 Nr. 2 BGB mit Kenntniserlangung beginnt und daher ggf. gem. § 199 Abs.4 BGB bis zu zehn Jahren dauern kann⁴²³.

Wird modifiziertes Kaufrecht angewandt, so sind Mängel ausgeschlossen, wenn der Softwarebesteller sie kennt (§ 442 Abs.1 BGB). Dies gilt auch bei Werkvertragsrecht gem. § 640 Abs.2 BGB. Während § 640 Abs.2 BGB aber auf die Abnahme abstellt, kommt es nach § 442 Abs.1 BGB auf den Zeitpunkt des Vertragsschlusses an⁴²⁴.

Im Handelsverkehr, also der Regelsituation für die hier untersuchten Verträge, müsste bei der Annahme von modifiziertem Kaufrecht zudem die Rügepflicht des § 377 HGB berücksichtigt werden⁴²⁵. Die Anwendbarkeit der Vorschriften des Handelskaufes ist jedoch keineswegs zwangsläufig, wenn der Vertrag dem Regime des § 651 S. 3 BGB unterstellt wird, da nach dem Gesetzeswortlaut die Vorschriften über den Kauf für anwendbar erklärt werden, es typologisch jedoch bei einem Werkvertrag bleibt.

Die Typologie des BGB geht in ihrer Systematik konsequent davon aus, bei (modifizierten) Kaufverträgen würde die Entscheidungsgrundlage für den

⁴²² Weick, Günter, Vereinbarte Standardbedingungen im deutschen und englischen Bauvertragsrecht, München, 1977, S. 188.

⁴²³ Diese Verjährung wird von Koch als „Drei-Plus-Verjährung mit erheblichem Risikopotential für den Softwareersteller bezeichnet, ITRB 2002, S. 297; Schmid will diese Konsequenz vermeiden, indem er für die Sacheigenschaft nach § 634a BGB einen anderen Maßstab als bei § 651 BGB anlegt, MMR 2004, S. 590; Thewalt hält die längere - verschuldensunabhängige - Verjährung im Falle der unkörperlichen Software als komplexes Gesamtwerk demgegenüber für angemessen, a.a.O., S. 256ff.

⁴²⁴ Dies führt für noch herzustellende Sachen zu der misslichen Situation, dass die Käufer ihre Rechte de facto nicht verlieren können, da es an einer Abnahme fehlt; sie wären also ungerechtfertigt besser gestellt als Besteller nach Werkvertragsrecht, Thewalt, a.a.O., S. 82.

⁴²⁵ Was etwa dazu führen könnte, dass die Ablieferung ohne geschuldetes Handbuch oder Bedienungsanleitung die Rügepflicht des Softwarebestellers in Gang setzt und er sie u.U. nach versäumter Anzeige nicht nachfordern kann, vgl. MüKo-HGB / Grunewald § 377 Rn. 24.

Vertragsschluss bereits vor diesem geschaffen. Der Käufer trifft seine Wahl aufgrund existenter oder hinreichend konkret bekannter Eigenschaften des Werkes. Beim Werkvertrag betrifft die Entscheidung dagegen lediglich eine Planung im Hinblick auf ein möglicherweise vage beschriebenes Werk. Im Ergebnis bestehen erhebliche Unterschiede für die Vertragspartner, je nachdem, ob eine Einordnung nach den §§ 651, 433ff oder 631ff BGB erfolgt.

4. In sonstigen Konstellationen („Lizenzvertrag“, §§ 705ff, F&E-Vertrag, „dynamischer Vertrag“)

Geht man von einem Lizenzvertrag aus, so soll die Summe aller jener Regeln Anwendung finden, „die sich aus den allgemeinen Bestimmungen des Schuldrechts auch mit Bezug auf den Lizenzvertrag ergeben oder aber durch die Rechtsprechung ermittelt werden“⁴²⁶. Soweit nicht allgemeine Regeln greifen, sei nach „passenden, das Wesen des Lizenzvertrages reflektierenden besonderen Regeln“⁴²⁷ zu suchen.

Soweit in der Literatur eine gesellschaftsrechtliche Konstruktion in Betracht gezogen wird⁴²⁸, wird dies mit einem gemeinsamen Erfolgsrisiko begründet⁴²⁹. Die Basis für eine solche Annahme bildet der gemeinsame Zweck⁴³⁰. Soweit eine gesellschaftsrechtliche Lösung überhaupt in Betracht gezogen wird, äußern sich hier bereits Zweifel⁴³¹. In ausführlicheren Untersuchungen wird die Anwendbarkeit der §§ 705ff deutlich abgelehnt. Die Parteien verfolgten nicht gemeinsame, sondern gegensätzliche Interessen⁴³², zudem mache es wertungsmäßig keinen Sinn, Vorschriften der §§ 705ff BGB auf Softwareentwicklungsverträge anzuwenden⁴³³. Ein gemeinsames

⁴²⁶ Hilty, MMR 2003, S. 3.

⁴²⁷ Hilty, MMR 2003, S. 3, beispielhaft für allgemeine Regeln werden die §§ 275ff, 311f, 313ff und 320 bis 326 genannt..

⁴²⁸ S.o. Fn. 352.

⁴²⁹ Karger, ITRB 2004, S. 208: liegt es bei beiden Parteien, spräche einiges für die Anwendung der §§ 705ff BGB.

⁴³⁰ Sobola/Dobmeier / Glöckler, a.a.O., S. 230.

⁴³¹ Karger etwa meint, die Anwendung von Gesellschaftsrecht werde aus diesem Grund (Erfordernis von gemeinsamem Zweck und hierauf gerichteten Fürsorgepflichten) ein Ausnahmefall sein, ITRB 2004, S. 208.

⁴³² Slongo, a.a.O., S. 142.

⁴³³ § 723 Abs.1 BGB beschränke die sonst gem. § 649 BGB jederzeitige Kündigungsmöglichkeit des Bestellers durch das Erfordernis eines wichtigen Grundes, Thewalt, a.a.O., S. 225f; das Individualrisiko der Beteiligten werde nicht notwendig reduziert, sondern durch eine Risikovergemeinschaftung ersetzt, Heppner, a.a.O., S. 308.

Interesse könne sich allenfalls ergeben, wenn Besteller und Softwarehaus das erstellte Produkt gemeinsam vermarkten wollten⁴³⁴.

Eine ähnliche Sonderrolle spielten Verträge, bei denen es sich um die Entwicklung „relativ neuer, völlig ungewöhnlicher Software in einer komplexen Umgebung“ handelt⁴³⁵. In diesem Fall sei von einem Entwicklungsauftrag auszugehen, für den das Werkvertragsrecht eine unangemessen einseitige Risikozuweisung zu Lasten des Softwarehauses enthalte⁴³⁶. Wenn der Arbeitserfolg maßgeblich von nicht unerheblicher Unsicherheit über das Erreichen des Zieles gekennzeichnet ist, müsse eine Lösung jenseits von Dienst- und Werkvertrag gefunden werden, bei dem im Falle des Fehlschlagens des Projektes ein Kündigungsrecht ex nunc und eine hälftige Kostentragung durch beide Vertragspartner die Folge sei⁴³⁷. Heppner schließlich stellt die Komplexität der Softwareentwicklung in den Vordergrund, bei der sich die Grenzen (austausch-) vertrags- bzw. gesellschaftsrechtlicher Konstruktionen zeige. Als Lösung biete sich ein dem Spiralmodell der Softwareentwicklung⁴³⁸ angenäherter sog. „evolutionärer Vertrag“ an⁴³⁹, der durch ein „wohlgeordnetes Ineinandergreifen von gesellschafts- und austauschvertraglichen Elementen“ gekennzeichnet sein müsse⁴⁴⁰.

IV. Leistungsbestimmung durch das „Pflichtenheft“

1. Begriff und Bedeutung

Unabhängig von der Einordnung des Vertragstyps spielt bei der Leistungsbestimmung häufig das sog. Pflichtenheft eine Rolle. Es stellt die Definition der zu erbringenden Leistung dar⁴⁴¹, wobei es nicht das „Wie“

⁴³⁴ Redeker, IT-Recht, a.a.O., S. 490.

⁴³⁵ Redeker, IT-Recht, a.a.O., S. 489.

⁴³⁶ Redeker, IT-Recht, a.a.O., S. 489.

⁴³⁷ Es handle sich um einen Typenverschmelzungsvertrag, der das Risiko weder ausschließlich dem Besteller (wie beim Dienstvertrag) noch ausschließlich dem Unternehmer (wie beim Werkvertrag) zuordne: Thewalt, a.a.O., S. 120ff; bei klar definierten Anforderungen fehle es aber an dem erheblichen Risikoelement, der Auftragnehmer gäbe mit der Akzeptanz eines Projektes mit klarer Definition der Aufgabenstellung konkludent zu erkennen, er wolle das Erfolgsrisiko übernehmen (S. 121f).

⁴³⁸ Vgl. dazu oben, Kapitel 1 S. 14ff.

⁴³⁹ Heppner, Softwareerstellungverträge, S. 192ff.

⁴⁴⁰ Heppner, a.a.O., S. 382.

⁴⁴¹ s.o. Kapitel 2, S. 20ff.

sondern das „Was“ enthält⁴⁴², also das Ziel des Projektes beschreibt, nicht jedoch den Weg dorthin. Aus technisch-organisatorischer Sicht existieren umfangreiche Definitionen⁴⁴³, die allerdings aus rechtlicher Perspektive bisher wenig Beachtung finden⁴⁴⁴. Auch aus dem Blickfeld des Softwareingenieurs steht die Fixierung der Kundenanforderungen in einem Pflichtendokument insbesondere für zukünftige Konflikte im Mittelpunkt des Interesses⁴⁴⁵.

Die Einschätzung der Bedeutsamkeit⁴⁴⁶ und Konsequenzen in der Projektpraxis fallen deutlich auseinander⁴⁴⁷. Anschaulich wird dieser Befund bei Betrachtung der bis in jüngste Zeit reichenden Judikatur, die Fälle zu beurteilen hatte, in denen ein Pflichtenheft schlicht nicht erstellt wurde⁴⁴⁸. Während also die Juristen vor allem aus einer ex post Betrachtung nicht müde werden, die Bedeutung sorgfältiger Anforderungsdefinitionen zu betonen, scheinen die am Projektgeschäft Beteiligten häufig auf eine entsprechende Fixierung zu verzichten.

⁴⁴² Koch, Frank A., Computervertragsrecht, 6. Aufl., Freiburg, 2002, Rn. 12, der auch von der vertraglichen Beschreibung des Lieferumfanges spricht; vgl. auch <http://de.wikipedia.org/wiki/Pflichtenheft> und [.../Software_Requirements_Specification](http://de.wikipedia.org/wiki/Software_Requirements_Specification).

⁴⁴³ So beispielsweise in der DIN 69901, die Begriffe des Projektmanagement definiert, oder in der Norm des Institute of Electrical and Electronic Engineers (IEEE) zur Software Requirement Specification (SRS in der Version Std 830-1998).

⁴⁴⁴ Eine Ausnahme ist beispielsweise aus jüngster Zeit der Beitrag von Söbbing (Die Bedeutung der ITIL für die Vertragsgestaltung, ITRB 2005, S. 97) zu der IT Infrastructure Library, kurz ITIL, wo der Autor ausdrücklich darauf hinweist, dass der Verweis auf technisch ausgelegte Referenzen das Risiko einer Änderung der Leistungsbeschreibung mit sich bringt, weshalb die Priorisierung der unterschiedlichen Dokumente zur Bestimmung der Leistungspflicht bedeutsam ist.

⁴⁴⁵ Mall, Rajib, Fundamentals of Software Engineering, New Delhi, 2003, S. 23: „*The SRS document (Anm.: Software requirement specification) normally serves as a contract between the development team and the customer. Any future dispute between the customer and the developer can be settled by examining the SRS document.*“; letztere Aussage dürfte nach den Erfahrungen der Rechtspraxis allerdings eher Wunschdenken sein.

⁴⁴⁶ Schaub betont die Bedeutung des Detaillierungsgrades für die Klarheit des Projektganges, S. 329; Ihde unterstreicht die Bedeutung des Pflichtenhefts als Maßstab für die zu erbringende Leistung, Ihde, Thomas, Das Pflichtenheft beim Softwareerstellungsvertrag, CR 1999, S. 409; Schneider weist auf das Risiko eines „Rundum-Sorglos-Pakets“ für das Softwarehaus vor dem Hintergrund unzureichender Leistungsbeschreibung hin, CR 2003, S. 317; auch in der Rechtsprechung wird die zentrale Bedeutung eines Pflichtenhefts für die Definition der Leistungspflicht unterstrichen: OLG Düsseldorf, U.v. 10.06.92, CR 1993, S. 361; OLG Köln, U.v. 03.12.93, OLGR Köln 1994, S. 13; OLG Köln, U.v. 12.02.99, NJW-RR 1999, S. 1733.

⁴⁴⁷ Siehe dazu bereits oben unter Kapitel 5. A. III. 2.

⁴⁴⁸ OLG Düsseldorf, U.v. 10.06.92, CR 1993, S. 361; OLG Köln; U.v. 26.06.92, OLGR Köln 1992, 237-238; OLG Köln, U.v. 18.06.93, OLGR Köln 1993, S. 237; OLG Köln, U.v. 25.06.93, OLGR Köln 1993, S. 270; OLG Köln, U.v. 03.12.93, OLGR Köln 1994, S. 13; LG Heilbronn U.v. 16.12.93, CR 1994, S. 281; OLG Saarbrücken, U.v. 22.09.94, BB Beilage 95, Nr. 16, S. 12; OLG Köln, U.v. 22.09.95, NJW 1996, S. 1067; OLG Düsseldorf, U.v. 18.07.97, CR 1997, S. 732; OLG Köln, U.v. 12.02.99, NJW-RR 1999, S. 1733; OLG Karlsruhe, U.v. 16.08.02, CR 2003, S. 95.

Wenn dem Pflichtenheft also einerseits eine große Bedeutung zuerkannt wird, es aber andererseits häufig nicht oder mangelhaft erstellt wird, stellt sich die Frage nach der Verantwortlichkeit für sein (Nicht-) Vorhandensein.

2. Verantwortungszuweisung

Zunächst ist klarzustellen, ob vertraglich zwischen der Planung und der Ausführung unterschieden wurde⁴⁴⁹. Wird eine Unterscheidung vorgenommen, so beschränkt sich der erste Teil des Vertrages auf die Erstellung eines Konzeptes für das weitere Vorgehen⁴⁵⁰. Hier liegt es nach einer Auffassung nahe, eine Parallele zum Baurecht zu ziehen⁴⁵¹. In der Planungsphase liege die Projektverantwortung beim Besteller, in der Erstellungsphase beim Softwarehaus⁴⁵². Aus diesem Grundsatz wird auch bei fehlender Phasentrennung der Ansatzpunkt für die Zuordnung der Verantwortlichkeit gefunden. Es sei „Aufgabe des Auftraggebers, klarzustellen, wie das von ihm gewünschte Werk auszusehen hat, also dessen Eigenschaften, Beschaffenheiten und Einsatzzwecke festzulegen und bekannt zu geben“⁴⁵³. Als „Konkretisierung der Aufgabenstellung und des Erwartungshorizonts“ sei die Erstellung des Anforderungsprofils originäre Pflicht des Auftraggebers⁴⁵⁴. Diese Aufgabenverteilung wird auch umgekehrt von Autoren aus dem Bereich des Baurechts für das Werkvertragsrecht betont: Die Planung falle grundsätzlich in den Aufgabenbereich des Auftraggebers⁴⁵⁵. Auch Thewalt sieht die Verantwortung zur Erstellung des Pflichtenhefts beim Auftraggeber und will bei Vorliegen eines Handelsgeschäftes aus § 375 HGB sogar eine Pflicht

⁴⁴⁹ Bereits Malzer hält bei der vorgelagerten „Erarbeitung eines Anforderungsprofils“ eine gesonderte rechtliche Beurteilung (d.h. nicht nach §§ 631ff BGB) für möglich, a.a.O., S. 266f, dies gelte insbesondere dann, wenn eine gesonderte Vergütung verlangt ist oder der Entwicklungsauftrag erkennbar noch nicht erteilt werden soll (S. 277).

⁴⁵⁰ Von Slongo als erfolgsgerichtete Tätigkeit im Interesse des Bestellers bezeichnet, bei der das Softwarehaus für kein bestimmtes Arbeitsergebnis einzustehen habe, a.a.O., S. 115.

⁴⁵¹ Redeker empfiehlt, das Erarbeiten eines Pflichtenhefts und die Erstellung selbst wie in den Leistungsphasen der HOAI sorgfältig zu trennen, IT-Recht, a.a.O., S. 306.

⁴⁵² Schneider, Handbuch, a.a.O., S. 1489.

⁴⁵³ Schaub, CR 1993, S. 329 unter ausdrücklicher Bezugnahme auf das Bau- und Werkmängelrecht.

⁴⁵⁴ Ihde, CR 1999, S. 409, der diese Zuordnung auch mit dem Argument unterstreicht, die Informationen kämen schließlich aus dem Verantwortungsbereich des Auftraggebers, dabei verweist er auf die Vergleichbarkeit mit dem Bau- und Industrieanlagenvertrag; Schneider, Handbuch, a.a.O., S. 1479.

⁴⁵⁵ Weick, in: Nicklisch/Weick, Verdingungsordnung für Bauleistungen, 3. Aufl., München, 2001, § 3 Rn.1, auch im Werkvertragsrecht gelte der Grundsatz, dass der Bauherr brauchbare und zuverlässige Pläne und Unterlagen zur Verfügung stellen muss (Rn.5).

dazu herleiten⁴⁵⁶. Malzer leitet die Verantwortlichkeit des Auftraggebers allgemein aus der Privatautonomie her, wonach „jede Partei primär selbst für die Einbringung ihrer Interessen Sorge zu tragen“ habe⁴⁵⁷.

Die Rechtsprechung verortet die Verantwortlichkeit für das Pflichtenheft ganz überwiegend ebenfalls beim Auftraggeber⁴⁵⁸. Es findet sich jedoch auch die Auffassung, beide Vertragspartner müssten an der Erstellung mitwirken, wobei der Auftragnehmer die Folgerungen aus der Ist-Analyse zu ziehen habe⁴⁵⁹. Wurde kein Pflichtenheft erstellt, so müsse der Auftragnehmer mit Änderungen und Ergänzungen rechnen und auf diese schnell reagieren⁴⁶⁰, und es sei ein mittlerer Ausführungsstandard geschuldet⁴⁶¹.

Dieses Ergebnis für die Verantwortungszuweisung ist unvollständig, soweit die damit zusammenhängenden Beratungspflichten des Softwarehauses außer acht gelassen werden. Auf diese ist daher näher einzugehen.

V. Beratungs- und Mitwirkungspflichten

Sowohl in der Literatur⁴⁶² als auch in der Rechtsprechung⁴⁶³ werden die intensiven Leistungsbeziehungen der Vertragspartner bei einem Softwareentwicklungsprojekt hervorgehoben. Heppner spricht hier

⁴⁵⁶ A.a.O., S. 141ff, dass nach § 375 HGB die nähere Bestimmung der (kaufvertraglichen) Verhältnisse dem Käufer *vorbehalten* sein muss, will Thewalt durch stillschweigende Vereinbarung eines Bestimmungsrechtes als gegeben ansehen (S. 149). Die (sich aus § 642 BGB ergebende) Obliegenheit zur Leistungsbestimmung enthalte als *minus* den Vorbehalt dazu (S. 150); ähnlich: Schneider, Handbuch, a.a.O., S. 1479, der die Erstellung des Pflichtenhefts als eine Mitwirkungsleistung des Auftraggebers verstanden wissen will.

⁴⁵⁷ Malzer, a.a.O., S. 277.

⁴⁵⁸ OLG Celle, U.v. 20.02.91, CR 1991, S. 610; OLG Köln, U.v. 18.06.93, OLGR Köln 1993, S. 237 und U.v. 25.06.93, OLGR Köln 1993, S. 270; LG Heilbronn, U.v. 16.12.93 CR 94, S. 281; OLG Saarbrücken, U.v. 22.09.94, BB Beilage 95, Nr. 16, S. 12; der BGH weist auf die Notwendigkeit hin, im Einzelfall zu ermitteln, ob die Erstellung eines Pflichtenhefts geschuldet sei, U.v. 13.11.97, NJW-RR 1998, S. 1128.

⁴⁵⁹ OLG Düsseldorf, U.v. 10.06.92, CR 1993, S. 361.

⁴⁶⁰ KG Berlin, U.v. 01.06.90, CR 1990, S. 768.

⁴⁶¹ OLG Düsseldorf, U.v. 18.07.97, CR 1997, S. 732 und OLG Frankfurt, U.v. 03.05.01, OLGR Frankfurt 2001, S. 307.

⁴⁶² Slongo, a.a.O., S. 31 spricht von einer „intensiven gesellschaftsähnlichen Zusammenarbeit“; nach Rehmann, Herstellung von Individualsoftware – Folgen der Verletzung von Mitwirkungspflichten, CR 1989, S. 961, ist die Werkleistung „*mit den Mitwirkungspflichten des Bestellers verflochten*“; vgl. auch Malzer, a.a.O., S. 288; Thewalt, a.a.O., S. 198; Sobola/Dobmeier/Glöckler, a.a.O., S. 227; Bormuth, Ulrich/Schmitz, Karl, Beteiligung der Anwender an der Softwareentwicklung, AiB 1993, S. 31; Nicklisch, Fritz, Kooperation bei Projektverträgen am Beispiel von Softwareentwicklungsverträgen, FS Traub, 1994, S. 305; Ihde, CR 1999, S. 409; Schubert/Haase, ZKM 2003, S. 251.

⁴⁶³ So bereits OLG Stuttgart, U.v. 18.10.88, CR 1989, S. 589 und zuletzt dem Inhalt nach auch BGH, U.v. 20.02.01, NJW 2001, S. 1718.

beispielsweise von „komplexen Zusammenhängen und symbiosehaften Abhängigkeiten“⁴⁶⁴ und Nicklisch von dem Erfordernis „intensiver Kooperation“⁴⁶⁵. Technik, Arbeitsorganisation und Qualifizierung der Mitarbeiter stellten zusammengehörende und nicht trennbare Teile eines integrierten Planungsprozesses dar⁴⁶⁶. Das Softwarehaus und der Besteller müssten „eng und kooperativ miteinander arbeiten“⁴⁶⁷. Daraus werden für die Pflichten der Beteiligten teilweise sehr weitgehende Konsequenzen gezogen.

1. Beratungs- und Dokumentationspflichten des Auftragnehmers

Bereits in früheren Entscheidungen pointiert die Rspr. die Pflichten des Softwarehauses bei der Individualsoftwareentwicklung. Die Entwicklung müsse in einem intensiven Dialog stattfinden, den der Auftragnehmer anzubieten und zu leiten habe⁴⁶⁸. Auch wenn die Pflichtenhefterstellung von Auftraggeberseite zu erfolgen habe, so treffe das Softwarehaus doch eine Pflicht zu umfangreicher und sorgfältiger Prüfung⁴⁶⁹. Die Aufgabenanalyse und Beratung sei, unabhängig von der Einordnung als Kauf- oder Werkvertrag, selbstverständlich und erforderlich⁴⁷⁰.

Diese Einschätzung wird von der älteren Literatur geteilt. Es stelle eine notwendige Hauptleistung des Softwarehauses dar, zur Betriebsorganisation zu beraten, die der Projektierung einer geeigneten Informatiklösung voran- bzw. mit ihr einhergeht⁴⁷¹. Der Umfang der Nebenpflichten des Softwarehauses hänge allerdings von dem Maß des Vertrauens auf seine Fachkunde ab⁴⁷². Dieses wiederum richte sich nach der eigenen Fachkunde des Auftraggebers⁴⁷³ und der Komplexität der Software sowie der von beiden

⁴⁶⁴ a.a.O., S. 174: dies führe vertrags- und gesellschaftsrechtliche Konstruktionen an ihre Grenzen.

⁴⁶⁵ a.a.O., S. 305.

⁴⁶⁶ Bormuth/Schmitz, AiB 1993, S. 31.

⁴⁶⁷ Müller-Hengstenberg, CR 2005, S. 385 (386), der dabei besonders die Phase des „Designs“ der Projektziele hervorhebt.

⁴⁶⁸ OLG Stuttgart, U.v. 01.10.88, CR 1989, S. 598.

⁴⁶⁹ OLG Celle, U.v.20.02.91, CR 1991, S. 610; OLG Köln, U.v. 18.06.93, OLGR Köln 1993, S. 237 und U.v.25.06.93, OLGR Köln 1993, S. 270.

⁴⁷⁰ BGH, U.v. 14.07.93, NJW 1993, 2436-2438.

⁴⁷¹ Dazu gehörten auch Rahmenorganisation, die Einführung der neuen Lösung im Betrieb des Anwenders sowie die notwendige Schulung des Personals Slongo, a.a.O. S.23, ebenso Malzer, a.a.O., S. 287.

⁴⁷² Malzer, a.a.O., S. 289.

⁴⁷³ Schaub, CR 1993, S. 329.

Seiten gestellten Fragen⁴⁷⁴. Teilweise wird in der neueren Literatur eine Prüfungspflicht des Softwarehauses bezüglich des Pflichtenhefts verneint⁴⁷⁵. Es wird dabei auch betont, das Softwarehaus habe zwar die Bedürfnisse des Bestellers sorgfältig zu analysieren und ggf. entsprechende Nachfragen zu stellen, jedoch dürfe dabei nicht unberücksichtigt bleiben, dass auf Seiten des Softwarehauses möglicherweise keine Branchenkenntnis, d.h. detailliertes Wissen über die Gegebenheiten in der Branche des Kunden, vorliege⁴⁷⁶.

Eine nicht selten streitentscheidende, softwaretypische Pflichtenerweiterung nehmen die Gerichte bezüglich der Dokumentation vor. Der Besteller habe Anspruch auf eine umfassende schriftliche Fixierung der Funktions- und Bedienungsregeln⁴⁷⁷. Die Lieferung gehöre auch ohne ausdrückliche Vereinbarung zur Hauptleistungspflicht⁴⁷⁸. Der BGH begrenzte diese extensiven Tendenzen etwas, indem er klarstellte, die Dokumentation müsse erst zum Abschluss der Arbeiten vorliegen und Teildokumentationen könnten nur bei entsprechender Absprache verlangt werden⁴⁷⁹.

2. Mitwirkungspflichten des Auftraggebers

Zunächst wird in der Literatur von einer Zuordnung der Mitwirkungspflichten zu den Obliegenheiten⁴⁸⁰ ausgegangen, d.h. ihre Verletzung habe zwar eine Rechtsverkürzung zur Folge, begründe jedoch keinen klagbaren Anspruch⁴⁸¹. Dies könne sich jedoch insbesondere durch eine Vereinbarung der Vertragsparteien ändern⁴⁸². Teilweise wird auch davon ausgegangen, auch ohne ausdrückliche Erwähnung im Vertrag stellten die Mitwirkungspflichten –

⁴⁷⁴ Redeker, IT-Recht, a.a.O., Rn. 420.

⁴⁷⁵ Schneider, Handbuch, a.a.O., S. 1538: Der Auftragnehmer habe keine genuine Prüfungspflicht, der Besteller müsse selbst für seine Angaben im Pflichtenheft einstehen, daher gingen Vagheit und Unvollständigkeit zu Lasten des Bestellers, wobei das Softwarehaus eine Hinweispflicht treffen kann, wenn es die Mangelhaftigkeit hätte erkennen können (S. 1539).

⁴⁷⁶ Redeker, IT-Recht, a.a.O., Rn. 420ff.

⁴⁷⁷ OLG Celle, U.v. 03.03.92, NJW-RR 1993, S. 432.

⁴⁷⁸ OLG Köln, U.v. 03.12.93, OLGR Köln 1994, S. 13; ebenso OLG Karlsruhe, U.v. 16.08.02, CR 2003, S. 95.

⁴⁷⁹ U.v. 20.02.01, NJW 2001, S. 1718.

⁴⁸⁰ Malzer, a.a.O., S. 290; Redeker, IT-Recht, a.a.O., Rn. 432; ausführlich, Thewalt, a.a.O., S. 208ff; Schneider, Handbuch, a.a.O., S. 1530.

⁴⁸¹ Esser/Schmidt, a.a.O., § 6 VI (S. 111ff).

⁴⁸² Thewalt, a.a.O., S. 229; Schneider, Handbuch, a.a.O., S. 1530; Redeker, IT-Recht, a.a.O., S. 434.

softwarevertragstypisch – echte Schuldnerpflichten dar⁴⁸³. Wird die Mitwirkung als Schuldnerpflicht verstanden, so kann das Softwarehaus aus einem etwaigen Schuldnerverzug Schadensersatzpflichten herleiten. Kommt man über § 651 S.3 BGB zur Anwendung des § 642 BGB, so kann das Softwarehaus demgegenüber (nur) eine angemessene Entschädigung, was den entgangenen Gewinn nicht einschließt, verlangen⁴⁸⁴. Um diesbezügliche Unklarheit zu vermeiden, wird eine deutliche Benennung der Mitwirkungspflicht empfohlen⁴⁸⁵.

Das Angewiesensein des Softwarehauses auf die Mitwirkung des Bestellers hat auch in der Rechtsprechung seinen Niederschlag gefunden. So erklärte der BGH die Gestellung kompetenter Mitarbeiter seitens des Bestellers für den erfolgreichen Abschluss des streitigen Projektes für erforderlich und sah in ihrem Fehlen eine schuldhaftige Mitverursachung der eingetretenen Schäden⁴⁸⁶. Es stelle auch eine Verletzung der Mitwirkungspflicht dar, wenn der Besteller das Pflichtenheft, als notwendige Voraussetzung für die zu erbringende Leistung, dem Softwarehaus auf Anforderung nicht vollständig überlasse⁴⁸⁷. Das LG Frankfurt erblickte eine solche in dem Nichtwahrnehmen der vom Softwarehaus angeforderten Besprechung⁴⁸⁸.

VI. Änderungen der Leistungsbestimmung

Welche Bedeutung Änderungen im Rahmen der Softwareentwicklung zukommt, bemerkt bereits Slongo auch aus rechtlicher Sicht⁴⁸⁹ prägnant: Projektänderungen gehörten zum „Wesen jedes nicht ganz einfachen Informatikprojektes“⁴⁹⁰. Ein Änderungsverfahren müsse daher vertraglich vorgesehen werden⁴⁹¹.

⁴⁸³ Rehmann, CR 1989, S. 961; auch nach dem BGH können Pflichten, denen nach dem Parteiwillen wesentliche Bedeutung zukommt, zu Hauptpflichten werden, U.v. 22.07.98, NJW 1998, S. 3197.

⁴⁸⁴ Vgl. zu den weiteren Rechtsfolgen ausführlich Thewalt, a.a.O., S. 199ff.

⁴⁸⁵ Sick, a.a.O., S. 31; Sobola/Dobmeier, a.a.O., S. 34.

⁴⁸⁶ BGH, U.v. 20.02.2001, NJW 2001, S. 1718, der Besteller befände sich dann in Annahmeverzug.

⁴⁸⁷ BGH, U.v.28.06.94, NJW-RR 1994, S. 1469 (1469).

⁴⁸⁸ LG Frankfurt, U.v.15.12.88, ECR-LG, S.79; weitere Beispiele aus der Rechtsprechung bei Koch, Computervertragsrecht, a.a.O., Rn. 230.

⁴⁸⁹ Die Bedeutung aus technischer Sicht wurde bereits in Kapitel zwei ausführlich dargestellt.

⁴⁹⁰ Slongo, a.a.O., S. 38.

⁴⁹¹ Redeker, IT-Recht, a.a.O., Rn. 437, im Zuge notwendiger Änderungen könne auch ein Festpreis unzumutbar werden, wobei - wie im Baurecht - die „Schmerzgrenze“ bei Budgetüberschreitungen von 20-25 % überschritten würde (S. 439).

Die Konsequenzen des Auftretens von Änderungen gewinnen in der Literatur allerdings weniger deutliche Konturen. Redeker stellt zunächst fest, es bestünde kein Anspruch des Bestellers auf Änderungen⁴⁹². Schneider sieht einen praktisch bedeutsamen Fall von Änderungen darin, dass Spezifikationen erst im Verlauf des Projektes vorgenommen würden. Konkretisiere der Besteller seine zunächst allgemeinen Vorgaben abschnittsweise, so verschwimme die Projektverantwortung⁴⁹³. Schneider bezeichnet es als „nichtadäquate Risikoverlagerung“, wenn der Besteller, der seine Vorgaben nicht sorgfältig erarbeitet, durch „kompensierende Pflichten des Auftragnehmers“ entlastet werde⁴⁹⁴.

Allgemeiner will Nicklisch danach unterscheiden, aus wessen Verantwortungsbereich die Projektstörungen kommen. Kämen sie aus dem des Softwarehauses, so käme weder eine zusätzliche Vergütung noch eine Fristverlängerung in Betracht. Stamme sie aus der Sphäre des Bestellers, so sei dagegen sowohl Fristverlängerung als auch Mehrvergütung geschuldet. Bei „neutraler“ Projektstörung müsste jeder seine Nachteile selbst tragen, die Fristen würden also verlängert, ohne dass allerdings eine zusätzliche Vergütung anfalle⁴⁹⁵.

Wie die Verantwortungsbereiche abgegrenzt werden können, bleibt dabei allerdings offen.

VII. UN-Kaufrecht (CISG)

Neben dem BGB bietet sich für Verträge mit internationaler Beteiligung das UN-Kaufrecht (CISG) als Rechtsquelle an⁴⁹⁶. Es ist von der Bundesrepublik Deutschland am 21.12.1989 ratifiziert worden und mit Wirkung zum 1.1.1991 in Kraft getreten. Da es sich um einheitliches *Kaufrecht* handelt, stellt sich zunächst die Frage, ob es auf Softwareentwicklungsverträge anwendbar ist.

⁴⁹² IT-Recht, a.a.O., S. 443, ebenso Nicklisch, FS Traub 1994, S. 305.

⁴⁹³ und mit ihr wanke die Einordnung als Werkvertrag, Schneider, Handbuch, a.a.O., S. 1485ff, 1487; die Pflicht des Softwarehauses zur Konkretisierung nicht geäußerter Vorgaben sei eine „Überforderung gegenüber dem Auftragnehmer“ (S. 1488).

⁴⁹⁴ Schneider, Handbuch, a.a.O., S. 1485.

⁴⁹⁵ Nicklisch, FS Traub 1994, S. 305.

⁴⁹⁶ Siehe hierzu bereits Kapitel 3, C II.

Auch im deutschen Recht wird die Diskussion über die Anwendbarkeit der CISG auf Softwareentwicklungsverträge im Rahmen des Art. 3 CISG geführt. Der Text der englischen Fassung⁴⁹⁷ lautet:

Article 3

1. *Contracts for the supply of goods to be manufactured or produced are to be considered sales unless the party who orders the goods undertakes to supply a substantial part of the materials necessary for such manufacture or production.*
2. *This Convention does not apply to contracts in which the preponderant part of the obligations of the party who furnishes the goods consists in the supply of labour or other services.*

Goods, die in der deutschen Fassung als „Waren“ übersetzt werden, sind in der CISG nicht definiert. Erfasst werden bewegliche Sachen im Gegensatz zu Rechten und Immobilien⁴⁹⁸. Die Sacheigenschaft von Software i.S des Art. 1 CISG ist nicht so umstritten wie bezüglich des BGB und die h.M. wendet das UN-Kaufrecht auf (Standard-) Software an⁴⁹⁹.

Bei den hier untersuchten Softwareentwicklungsverträgen ist weiter davon auszugehen, dass der Besteller zwar intensiv mitarbeitet, bzw. dazu verpflichtet wäre, er jedoch kein Vorprodukt wie etwa eine vorhandene Software zur Verfügung stellt⁵⁰⁰. Nach Art. 3 Abs.1 CISG wäre danach eine Anwendbarkeit gegeben.

⁴⁹⁷ Zu den offiziellen Gesetzessprachen gehören (gleichrangig) Arabisch, Chinesisch, Englisch, Französisch, Russisch und Spanisch, nicht jedoch die deutsche Übersetzung, die daher lediglich eine unverbindliche Anwendungshilfe darstellt, Ferrari, in: Schlechtriem, Peter (Hrsg.), Kommentar zum einheitlichen UN-Kaufrecht, 3. Aufl., München, 2000, Vor Artt. 1-6 Rn. 26f.

⁴⁹⁸ Staudinger/Magnus, Art. 1 Rn. 42; Schlechtriem/Ferrari, Art. 1 Rn. 34; Soergel/Lüderitz/Fenge, Art. 1 Rn. 21.

⁴⁹⁹ Staudinger/Magnus, Art. 1 Rn. 44 m.w.Nachw.; Soergel/Lüderitz/Fenge, Art. 1 Rn. 21, der sich auf die textliche Erweiterung des Art. 1 CISG gegenüber dem EKG (Das Einheitliche Gesetz über den internationalen Kauf beweglicher Sachen – als Vorläufer der CISG sprach von „objets mobiliers corporels“) bezieht; Endler, Maximilian/Daub, Jan, Internationale Softwareüberlassung und UN-Kaufrecht, CR 1993, S. 601 (603f); die eine weite Auslegung des Warenbegriffes fordern und auch bei unkörperlicher Verschaffung letztlich dasselbe Gut in den Mittelpunkt der Betrachtung stellen; differenzierend wie aus dem zum BGB bekannten Streit: Schlechtriem/Ferrari, Art. 1, Rn. 38, der Programme, die über einen Träger nutzbar sind, als Waren ansieht.

⁵⁰⁰ Bestünde der Auftrag des Softwarehauses in der Änderung vorhandener Software, so fände die CISG nach Art. 3 Abs.1 dann keine Anwendung, wenn diese Software einen *substantial part* ausmacht. Dies wird nach der überwiegenden Auffassung durch das Wertverhältnis des von den Parteien beizutragenden Materials bestimmt, Staudinger/Magnus, Art.3 CISG Rn. 14; ebenso Soergel/Lüderitz/Fenge, Art. 3, Rn. 3; a.A. Schlechtriem/Ferrari, der auch ein qualitatives Element berücksichtigen will, Art. 3 Rn. 8.

Für Individualsoftware hängt das Ergebnis allerdings weiter davon ab, ob die Pflicht der liefernden Partei, hier das Softwarehaus, zum überwiegenden Teil (*preponderant part*) aus Arbeits- oder sonstiger Dienstleistung (*supply of labour or other services*) besteht. Dies wird für Softwareentwicklungsverträge überwiegend bejaht⁵⁰¹.

Damit spielt das UN-Kaufrecht für Softwareentwicklungsverträge keine Rolle.

C. Zusammenfassung und Ergebnis

Für die rechtlichen Rahmenbedingungen im deutschen Recht ist als erster Grundsatz festzuhalten, dass geschriebenes Recht in Form der Gesetze den Ausgangspunkt rechtlicher Bewertung, die Referenz in Zweifelsfällen und den allein verbindlichen Rahmen für die gerichtliche Beurteilung darstellt. Auch die obergerichtliche Rechtsprechung entfaltet keine Bindungswirkung, wobei in der Praxis gleichwohl die faktische Orientierung vor allem an den höchstgerichtlichen Entscheidungen eine maßgebliche Wirkung ausübt. Eine Grundlage für das deutsche Vertragsrecht bildet die Privatautonomie, also das Recht der Vertragspartner, autonom und damit prinzipiell unberührt von staatlicher Einflussnahme über ihre privaten Verhältnisse zu bestimmen. Die Ausnahmen von diesem Prinzip spielen für den Untersuchungsgegenstand keine große Rolle. Dem entspricht die Konzeption des Schuldrechts als ein weitgehend offenes System. Der großen Bedeutung der Parteivereinbarung trägt das Gesetz durch Auslegungsvorgaben Rechnung. Maßgeblich ist der wirkliche Wille (§ 133 BGB) und der hypothetische Parteiwille, wie er sich aus dem Erklärten und aus den Begleitumständen (§ 157 BGB) entnehmen lässt. Das Gericht legt die Erklärungen der Parteien bei Vertragsschluss und die Umstände, unter denen diese abgegeben wurden, sowie die Vorgeschichte im Prozess ohne Bindung an den (vor allem im Prozess geäußerten) Parteiwillen aus. Dabei ist der Wortlaut ein Anhaltspunkt, der im Lichte des Empfängerhorizontes und der konkreten Umstände im Einzelfall zu prüfen ist. Der geäußerte Wille

⁵⁰¹ Staudinger/Magnus, Art. 1 Rn. 44: die wesentliche Vertragsleistung bestehe nicht in der Übertragung, sondern der Entwicklung (dem „Schreiben“) des Programms; Schlechtriem/Ferrari, Art. 1 Rn. 38; wohl auch Soergel/Lüderitz/Fenge, Art. 1 Rn. 21; Endler/Daub, die zwar den zur Produktion selbst erforderlichen Aufwand nicht zu den Dienstleistungen nach Art. 3 Abs.2 CISG rechnen, wohl aber die neben die Erstellungspflicht tretenden Installations-, Anpassungs-, Support-, Programmpflege- und Einarbeitungsarbeiten, a.a.O., S. 605.

hat gegenüber der (gerichtlichen) Auslegung infolge der Privatautonomie allerdings Vorrang.

Unabhängig vom Parteiwillen sind die Grundsätze von Treu und Glauben, Verkehrssitte sowie Handelsbräuche zu beachten, wobei deren Konturen eher mit einer Bürste als mit einem Pinsel gezogen werden. Die Folgen von Unklarheiten können durch die Gerichte im Konfliktfall anhand unbestimmter Rechtsbegriffe und weiten Ermessens der Richter bei der Berücksichtigung vertragsbegleitender Umstände in großem Umfang „gerichtsautonom“ bestimmt werden.

Bei der Frage nach dem konkret auf Softwareentwicklungsverträge anzuwendenden Recht ist in der Fachliteratur nahezu alles umstritten. Dies gilt zunächst für die Frage, ob ein Vertragstyp aus dem besonderen Schuldrecht zugeordnet werden muss. Es setzt sich bei der Frage danach fort, ob überhaupt ein existenter Gesetzestypus passt, und mündet anschließend in die Frage, um welchen Typus es sich handelt. Im Ergebnis wird von zwei breiten Strömungen mit unterschiedlichen Begründungen die Anwendbarkeit des (reinen) Werkvertragsrechts sowie des nach § 651 BGB modifizierten Kaufvertragsrechts vertreten. Daneben finden sich auch Ansichten, die keinen Typus des besonderen Schuldrechts oder eine Mischform als gegeben sehen.

Die Rechtsprechung verfolgt demgegenüber zur Rechtslage vor der Schuldrechtsreform eine klare Linie, wonach Softwareentwicklungsverträge nach Werkvertragsrecht zu beurteilen sind.

Einen zentralen Gesichtspunkt stellt in Literatur und Rechtsprechung die Frage nach der Sacheigenschaft von Software dar. In der Literatur wird dies ohne absehbare Klärung kontrovers diskutiert. Dabei findet häufig zunächst eine begriffliche Unterscheidung zwischen dem Programm und der Programmverkörperung statt, und anhand der Auffassung zur Trennbarkeit wird über die Einordnung als Sache entschieden. Die Rechtsprechung ordnet Software unbeirrt von der Diskussion in der Literatur unter den Sachbegriff ein. Ein besonderer Status wird der Software gleichwohl von einigen Stimmen in Rechtsprechung und Literatur hinsichtlich des auf sie anwendbaren Fehlerbegriffes zugestanden.

Entgegen der weitläufig betonten Bedeutung konkreter Leistungsbestimmung wird diese in der Praxis häufig gar nicht, verspätet und/oder mit erheblichen Abweichungen zur anfänglichen Konzeption vorgenommen. Die gesetzlichen Regelungen der Leistungsbestimmung sind in hohem Maße von der Einordnung unter den Vertragstypus abhängig. Hinsichtlich der Unterscheidung von Werk- und modifiziertem Kaufvertragsrecht zeigt sich dies insbesondere anhand der Differenzen von Abnahme (beim Werkvertrag) und Übergabe (beim Kaufvertrag). Auch hier spielt die Frage nach der Sacheigenschaft von Software im Rahmen des § 651 BGB wieder eine wichtige Rolle. Wird ein Lizenzvertrag angenommen, greifen allgemeine, kaum näher konturierte Regeln. Die Annahme einer Zuordnung von Elementen aus dem Gesellschafts- oder Dienstvertragsrecht wird an das seltene Vorliegen besonderer Konstellationen geknüpft. Eine „gemischte“ Lösung führt in der Konsequenz zu ebenso gemischten rechtlichen Rahmenbedingungen.

Die genaue Definition der zu erbringenden Leistung in einem sog. Pflichtenheft wird v.a. in der Literatur nachdrücklich gefordert, in der Praxis aber häufig nicht vorgenommen. Verantwortlich für die Erstellung des Pflichtenhefts ist nach ganz überwiegender Auffassung der Besteller der Software. Dabei müssen allerdings die sehr intensiven Leistungsbeziehungen zwischen den Vertragsparteien bei einem Softwareentwicklungsvertrag berücksichtigt werden, die zu erheblichen gegenseitigen Abhängigkeiten bei der Leistungserbringung führen. Das Softwarehaus trifft diesbezüglich vor allem eine, vom Informationsgefälle abhängige, Pflicht zu umfänglicher Beratung und Dokumentation. Es ist eine wichtige Aufgabe des Bestellers, durch geeignete Mitwirkung zum Projekterfolg beizutragen.

Häufig notwendige und/oder gewünschte Änderungen der Leistungsbestimmung können nach einer Auffassung in der Literatur vom Besteller regelmäßig nicht verlangt werden. Die Auswirkungen daraus resultierender Projektstörungen richten sich danach, aus wessen Verantwortungsbereich sie stammen. Es wird daher eine vertragliche Regelung des Vorgehens bei Änderungen empfohlen.

Dem UN-Kaufrecht kommt bei Softwareentwicklungsverträgen im deutschen Recht bisher keine relevante Rolle zu, da seine Anwendbarkeit an der überwiegenden Arbeits- bzw. Dienstleistung im Rahmen der Softwareentwicklung scheitert.

Kapitel 6: Empirischer Befund

In diesem Kapitel werden entsprechend Kapitel 4 neun Musterentwürfe und drei Beispiele aus der Praxis ausgewertet.

A. Musterentwürfe

Untersucht wurden die Musterentwürfe von Koch (KO)⁵⁰², Marly (MA)⁵⁰³, Bartsch (BA)⁵⁰⁴, Glöckler (GL)⁵⁰⁵, Schneider (SN)⁵⁰⁶, Schröder (SÖ)⁵⁰⁷, Heppner (HE)⁵⁰⁸ sowie die BVB-Erstellung (BVB)⁵⁰⁹. Nur Schneider unterscheidet zwischen einer am Besteller und am Softwarehaus orientierten Variante. Sie werden hier als SN-B und SN-S unterschieden. Mit Ausnahme von HE und MA gliedern sich alle Muster in einen Hauptvertrag und zum Teil zahlreiche Anlagen. Details bezüglich der zu erbringenden Leistung werden dabei in eine sehr unterschiedlich als „*Leistungsschein*“ (KO), „*Funktionsbeschreibung*“ (SÖ), „*Pflichtenheft*“ (MA und SN-B), „*Erstellungsschein*“ (GL und die BVB) oder „*fachliche Feinspezifikation*“ (SE-AN) bezeichnete Anlage verwiesen.

⁵⁰² Koch, Computervertragsrecht, Formular Nr. 7 im Anhang: „Vertrag über die Erstellung von Software“.

⁵⁰³ Marly, Softwareüberlassungsverträge, Formular VII in Teil F, „Überlassung von Individualsoftware im Unternehmensverkehr mit Quellcodeüberlassung, S. 708ff.

⁵⁰⁴ Münchener Vertragshandbuch / Bartsch, Bd. 3, Formular III. H.4, „Vertrag über ein Softwareprojekt“, S. 573ff.

⁵⁰⁵ Sobola/Dobmeier Glöcker, Vertragsmuster D III, „Softwareerstellungsvertrag“, S. 233ff.

⁵⁰⁶ Schneider, Handbuch, Vertragsbeispiele IV. und V. im Anhang, „Vereinbarung zur Softwareerstellung“ und „Vereinbarung zur Erstellung und Installation von Software“, S. 2380ff / 2396ff.

⁵⁰⁷ Schröder, Softwareverträge, Textabdruck, „Softwarelizenzvertrag“, S. 15ff.

⁵⁰⁸ Heppner, Softwareerstellungsverträge, Muster A1 im Anhang, „Muster für einen evolutionären Softwarevertrag zwischen Endanwender (E) und Softwareunternehmen (S)“ S. 387ff.

⁵⁰⁹ Entnommen aus: Müller-Hengstenberg, BVB-EVB-IT-Computersoftware, BVB-Erstellung, S. 180ff. Bei den Besonderen Vertragsbedingungen für die Beschaffung von DV-Leistungen (BVB) und den ergänzenden Vertragsbedingungen für die Beschaffung von IT-Leistungen (EVB-IT) handelt es sich um im Bundesanzeiger veröffentlichte Richtlinien, die von den Herstellerverbänden der Softwareindustrie unter Federführung des Bundesministers des Inneren seit 1972 erarbeitet wurden. Ihr Inhalt ist nicht unumstritten und sie werden daher vom BGH, im Gegensatz etwa zu der VOB/B, nicht als Standard anerkannt und der AGB-Kontrolle im Einzelfall unterzogen, vgl. BGH, in: CR 1991, S. 273 (274f).

I. Vertragsart

Auch die Wahl der Vertragsbezeichnung offenbart ein breitgefächertes Spektrum der Nomenklatur, wobei sich daraus teilweise Anhaltspunkte für den systematischen Ansatz des Autors ableiten lassen. KO („*Vertrag über Erstellung von Software*“), GL („*Softwareentwicklungsvertrag*“) und SN-S („*Vereinbarung zur Softwareentwicklung*“) verwenden sehr ähnliche Bezeichnungen. In SN-B wird dies in Andeutung einer Pflichtenerweiterung abgeändert auf „*Vereinbarung zur Erstellung und Installation von Software*“.

MA betitelt sein Vertragsmuster als „*Überlassung von Individualsoftware im Unternehmerverkehr mit Quellcodeüberlassung*.“ Die für maßgeblich gehaltenen Einteilungskriterien sind hier also einmal Individualsoftware (in Abgrenzung zur Standardsoftware), im „*Unternehmerverkehr*“ (in Abgrenzung zu Verträgen mit Verbrauchern) sowie die Überlassung des Quellcode (in Abgrenzung zur Überlassung nur im Objektcode, d.h. ohne die Weitergabe des in der Software verkörperten Know-How, als Voraussetzung für eine Änderung des Programms durch den Besteller oder von ihm beauftragte Dritte). Die Wortwahl „*Überlassung*“ (statt z.B. Erstellung) schließlich weist MA als Vertreter der Ansicht aus, die Softwareentwicklungsverträge dem (modifizierten) Kaufrecht zuordnet.

BA wählt die offene Bezeichnung „*Vertrag über ein Softwareprojekt*“, SÖ gibt durch den Titel „*Softwarelizenzvertrag*“ zu erkennen, dass er den Schwerpunkt der vertraglichen Regelung in lizenzrechtlichen Fragen sieht. HE schließlich bezeichnet seinen Entwurf als „*Muster für einen Evolutionären Software-Vertrag zwischen Endanwender und Softwareunternehmen*“⁵¹⁰. Die Besonderheit seines Ansatzes ergibt sich hier aus der Betonung des evolutionären Elements der Vereinbarung. Die Überschreibung der BVB als „*Besondere Vertragsbedingungen – Erstellung*“ schließlich läßt erkennen, dass es sich um klassische allgemeine Vertragsbedingungen i.S. des BGB handelt, die für einen bestimmten Anwendungsbereich, nämlich die Erstellung (in Abgrenzung zu Planung und Pflege, für die unterschiedliche BVB existieren), vorgesehen sind.

⁵¹⁰ Die von HE vorgestellte Variante „*Muster... zwischen zwei Softwareunternehmen*“ enthält keine für den Untersuchungsgegenstand durchgreifenden Unterschiede.

Alle Muster sehen eine Übertragung der Rechte an der Software vor, wobei MA (§ 7 Abs.2) und SN-B (Nr. 4.1, 4.4) eine exklusive Einräumung der Nutzungsrechte an den Softwarebesteller enthalten, während BA (§§ 27,28), SN-S (Nr. 4.1) und KO (§ 3) sich auf eine einfache Rechteübertragung beschränken. Als Standard mit abweichender Option ist dies auch bei HE (§ 10) der Fall. GL, SÖ und die BVB treffen keine Vorentscheidung, sondern überlassen die Wahl dem Musterverwender.

SN-B, GL und SÖ verwenden eine Präambel, wobei sich bei SN-B und GL bedeutsame Informationen finden.

Unter SN-B wird formuliert:

„Die Vertragspartner arbeiten seit längerer Zeit zusammen. Der Auftragnehmer (AN) hat für den Auftraggeber (AG) bereits Untersuchungen im Haus des AG durchgeführt und inzwischen abgeschlossen. Diese Ist-Analyse (vom...) bildet den Ausgangspunkt für die folgenden Regelungen. Der AN kennt aus der bisherigen Arbeit nicht nur die Verhältnisse bei dem AG, und zwar auch dessen EDV- und Organisationsverhältnisse, aus eigener Erfahrung, sondern verfügt auch über die entsprechenden Unterlagen.

Die Vertragspartner haben in der Folgezeit nach dieser Analyse eine Reihe von Projektsitzungen abgehalten, die in den Protokollen 1/02, 5/02 und 7/02 festgehalten sind. Darin sind auch die Empfehlungen des AN für das weitere Vorgehen und die Auswahl der Ziel-Hardware bzw. des Ziel-Betriebssystems enthalten. Auf der Basis dieser Gespräche steht auch fest, welche Ziele der AG mit seiner neuen Software verfolgt und welche strategischen Ziele hierdurch wieder realisiert werden sollen (z.B. produktionssynchrone Belieferung(...)).

Die Arbeiten des AN haben zu einem Pflichtenheft in Verbindung mit einem Terminplan mit relativen Terminen als Ergebnis geführt. Der AN hat in der Zwischenzeit ein Auswahlverfahren für die geeignete Hardware und Betriebssysteme sowie Datenbankumgebung durchgeführt und eine entsprechende Auswahl bereits getroffen. Diese Maßnahmen sind Grundlage dieses Vertrages.(...)“

Werden diese Ausführungen⁵¹¹ v.a. im Lichte der Kenntnis der rechtlichen Rahmenbedingungen aus Kapitel 5 näher betrachtet, so wird deutlich, warum die Präambel in SN-S nicht enthalten ist: Das Softwarehaus erkennt hier an, vollständig über die Gegebenheiten und Ziele des AG informiert zu sein. Das (hohe) Risiko, eine technische oder betriebliche Rahmenbedingung übersehen oder unzutreffend bewertet zu haben, liegt

⁵¹¹ Die Allgemeinheit der Formulierung ließe es zu, diese Präambel mit geringfügigen Änderungen in einer Vielzahl von Fällen aufzunehmen.

damit alleine beim Softwarehaus. Es wird mit dem Vortrag, etwas nicht gewusst zu haben, im von Dritten beurteilten Streitfall kaum Gehör finden.

Die Präambel bei GL hat einen „neutraleren“ Inhalt:

1.1. Anbieter

Der Anbieter ist eine Softwarefirma, die sich spezialisiert hat auf die Entwicklung von Software zur Lagerverwaltung. Sie verfügt über Erfahrungen mit Konzeption und Realisierung von EDV-Lösungen.

1.2. Kunde

Der Kunde ist ein mittelständisches Unternehmen, das mit Kfz-Zubehörteilen handelt. (...) Der Kunde verfügt über Erfahrung im Umgang mit Software zur Lagerverwaltung und genaue Kenntnis darüber, welche betrieblichen Prozesse durch die Lagerverwaltung erledigt werden sollen.“

Der Kunde bestätigt, dass er kein EDV-Laie ist und genau weiß, was er will. Das Softwarehaus bestätigt demgegenüber (nur), dass es über EDV-Fachkenntnisse verfügt, jedoch weder, dass es die betrieblichen Umstände beim Kunden noch dessen Intentionen genauer kennt. Es hätte daher deutlich geringere Schwierigkeiten damit, bei einer Auseinandersetzung darauf zu verweisen, ein Umstand sei ihm nicht bekannt gewesen und habe auch nicht bekannt sein müssen.

Was im Ergebnis geschuldet wird, beschreiben die Muster sehr unterschiedlich. HE betont in „§ 1 **Entwicklungsziel**“ die Gemeinsamkeit des Entwicklungsprozesses:

Das Ziel der Zusammenarbeit ist der gemeinsame Aufbau eines Management-Informationssystems (MIS), das sich an den technischen und wirtschaftlichen Anforderungen von E (Anm.: dem Softwarebesteller) sowie an den Qualitätsstandards des Weltmarktes orientiert.

SÖ formuliert in § 2, überschrieben mit „**Austauschvertrag**“:

„Mit dem vorliegenden Vertrag werden Leistungen ausgetauscht. Eine gesellschaftsrechtliche Verbindung zwischen dem Lizenzgeber und dem Lizenznehmer wird hierdurch nicht begründet.“

Hier klingt das Element gemeinsamer Verantwortung ebenfalls an.

BA und MA belassen es bei der Bestimmung des Vertragsgegenstands bei einer im Hinblick auf den Erfolg allgemeineren Formulierung. Nach BA, „§ 1 Projektvertrag“:

Der AG beauftragt hiermit den AN, das in diesem Vertrag umschriebene Softwareprojekt durchzuführen. Die Leistungen sind insbesondere:
a) *Lieferung von Standardsoftware;*
b) *umfassende Parametrisierung, Änderung und Erweiterung dieser Software;*
c) *Tests, Installation und Schulung.*
Leistungsziel ist, dass die Software getestet und betriebsfertig auf der Hardware des AG installiert ist.

In „§ 1 Vertragsgegenstand“ heißt es bei MA:

„Gegenstand des Vertrages ist das vom Hersteller zu entwickelnde und dem Anwender zu überlassende Computerprogramm „XY-Software“ einschließlich Benutzungsanleitung, Quellcode, Herstellerdokumentation sowie weiterer schriftlicher Materialien zur Produktbeschreibung, nachfolgend zusammenfassend als Software bezeichnet.“

GL, BVB und SN-S nehmen bereits im Hinblick auf die Beschreibung des Vertragsgegenstands ausdrücklich Bezug auf die konkreten Anforderungen des Softwarebestellers:

GL, „1.3 **Vertragsziel**“

Ziel des Austauschvertrages ist es, gegen Geldleistung eine speziell auf die betrieblichen Anforderungen des Kunden im Werk Lieferstraße 1, 11111 Handelshausen zugeschnittene, auf der derzeitigen Hardware des Kunden laufende EDV-Lösung zur Lagerverwaltung zu erarbeiten und einzuführen.“

BVB, „§ 3 **Leistungen des Auftragnehmers**“

1. Der Auftragnehmer ist verpflichtet, nach Maßgabe des Erstellungsscheines unter Ausnutzung des Standes der Wissenschaft und Technik bei Vertragsabschluss das DV-technische Feinkonzept, funktionsfähige Programme einschließlich Dokumentation zu erstellen sowie vereinbarte sonstige Leistungen zu erbringen.
Wenn der Auftragnehmer erkennt, dass die Leistungsbeschreibung (Ziffer 1 bis 3 des Erstellungsscheines) oder eine Forderung des Auftraggebers zur Vertragsausführung fehlerhaft, unvollständig, nicht eindeutig oder objektiv nicht ausführbar ist, hat er dies und die ihm erkennbaren Folgen dem Auftraggeber unverzüglich schriftlich mitzuteilen. Der Auftraggeber wird seinerseits unverzüglich über eine Änderung der Leistungsbeschreibung (§

5) oder seiner Forderungen zur Vertragsausführung (Nummer 3) entscheiden. (...)

Hier findet sich der Versuch, einen objektivierbaren Leistungsmaßstab einzuführen (*Stand der Wissenschaft und Technik bei Vertragsabschluss*). Zudem werden die Sorgfaltspflichten des Softwarehauses im Hinblick auf die Prüfung der Leistungsbeschreibung des Softwarebesteller konkretisiert.

SN-S, „1. Vertragsgegenstand“

„1.1 Der Auftragnehmer erstellt für den Auftraggeber Individual-Software (Vertragssoftware), deren fachliche Spezifikation vom Auftraggeber schon in der Ausarbeitung vom..., Autor..., Version..., niedergelegt ist. Diese fachliche Feinspezifikation beschreibt richtig, vollständig und abschließend den Leistungsumfang der vom Auftraggeber zu erstellenden Software. Sie ist Vertragsbestandteil und als Anlage 1 beigefügt. (...)

Durch diese Formulierung liegt die Verantwortung für die Spezifikation beim Softwarebesteller, und es wird der Versuch unternommen, vor Beginn der Arbeiten des Softwarehauses einen verlässlichen („richtig, vollständig und abschließend“) status quo zu schaffen.

Eine (noch) weiter konkretisierende Variante findet sich bei SN-B unter „1. Vertragsgegenstand.“:

1.1. Software-Erstellung

Der AN entwickelt und erstellt für den AG die Software XYZ gemäß dem Pflichtenheft, s.a. Präambel, sowie gemäß Terminplan, die beide Bestandteil dieses Vertrages sind (Anlage 1 und 2).

1.2. Projektverantwortung

1.2.1 Der AN übernimmt die Projektverantwortung auch für dieses Software-Produkt, hat sie jedoch schon im Rahmen des Gesamtprojektes inne und übernimmt die Koordination und Integration dieses Projektes in das Gesamtsystem.

1.2.2 (Mitwirkung des AG...)

1.2.3 Die vom AN erstellte Projektschätzung ist als Anlage 4 Vertragsbestandteil und Referenz für Minderungen und Mehrungen sowie Änderungen und deren Aufwand (s.a. Ziffer 2.4.6). (Anm.: Siehe dazu unten, VI. Veränderungen)

1.3. Geschuldetes Ergebnis

1.3.1. Das vom AN geschuldete Ergebnis ist die rechtzeitige Implementierung und Herstellung der Ablauffähigkeit der zu erstellenden Software mit einwandfreier Funktionsweise in abnahmefähigem Zustand

*innerhalb des Gesamtsystems mit Echtdateien bis spätestens zum ... Die Teilschritte ergeben sich aus dem Terminplan.
1.3.2. (Dokumentationen...)*

1.4. Berücksichtigung Gesamtprojekt

Der AN wird bei der Ausführung des Vertrages die Entwicklung des Gesamtprojektes sowie die technische Entwicklung berücksichtigen und den AG rechtzeitig auf etwa erforderlich werdende Änderungen hinweisen und dazu erforderlich werdende Entscheidungen herbeiführen. Soweit der AN dies unterlässt, schuldet er dennoch zum Gesamtpreis (s. Ziffer 8) eine entsprechend angepasste und/oder aktualisierte Leistung (nicht nur mittleren Ausführungsstandard).

Das Softwarehaus wird hier über die aus dem Pflichtenheft herzuleitende Leistungsbestimmung hinaus für die „Entwicklung“ von Gesamtprojekt und (außervertraglicher) Technik in die Verantwortung genommen. Es wird darüber hinaus eng an seine Prognosen insbesondere hinsichtlich Zeit und Kosten gebunden, und dies unbeschadet der vorgesehenen Festpreisvereinbarung.

Eine kürzere, aber im Ergebnis weitergehende Definition schlägt schließlich KO in „**§ 1 Vertragsgegenstand**“ vor:

Entwicklung

Der Anbieter entwickelt für das vom Kunden vorgegebene Problem eine zweckmäßige und wirtschaftliche EDV-Lösung in der Form geeigneter Software über die Stufen Problemanalyse/Systemplanung/Programmierung/Dokumentation und Einweisung. (...)

Das Softwarehaus soll hier dazu verpflichtet werden, ein „Problem“ des Softwarebestellers zu lösen, und zwar in einer „zweckmäßigen und wirtschaftlichen“ Weise mit „geeigneten“ Mitteln. Umfassender und allgemeiner könnte ein Pflichtenprogramm kaum definiert werden.

II. Bestandteile der Lieferung

Die Muster KO, MA, BA, GL, SN-S und SÖ sehen die Lieferung der Software vor, wobei die Bezeichnung divergiert:

KO unter 2.1: „*Programm in kodierter und eingabebereiter Form*“;

MA unter § 1: „*Computerprogramm*“;

BA unter § 1 (1): „*Standardsoftware*“⁵¹²;

GL unter 4.1 (2): „*Lizenzprodukt in der vereinbarten Code-Form gemäß VB 9* (Anm.: VB = Vertragsbedingungen, hier im Unterschied zu ES = Erstellungsschein verwendet), sowie unter 9.1: „*das Lizenzprodukt auf installationsfertigen Datenträgern*“.

SN-S unter 1.2: „*Software auf einem üblichen Datenträger*“;

SÖ unter § 7 (5): „*Lieferkopie des Lizenzproduktes auf Datenträgern*“.

Die BVB sprechen unter § 9 Nr.1 allgemeiner von der Pflicht des Auftraggebers, „*die vertraglichen Leistungen in der vereinbarten Form zu dem im Erstellungsschein festgelegten Zeitpunkt*“ zu übergeben. HE enthält keinerlei Lieferverpflichtung und SN-B statuiert - hierzu konträr - das umfassende Recht des Softwarebestellers „*die Herausgabe sämtlicher Arbeitsergebnisse und Unterlagen*“ zu verlangen (Nr. 4.3.6).

Hinsichtlich der zu liefernden Dokumente finden sich sowohl terminologisch als auch inhaltlich beträchtliche Unterschiede:

SÖ verlangt kurz die Aushändigung der „*Dokumentation in zweifacher Ausführung*“ (§ 7 (5)). KO sieht in § 2 Nr. 2.3 die Lieferung einer „*Benutzerdokumentation*“ sowie optional einer „*Entwicklerdokumentation*“, MA in § 1 die von „*Benutzungsanleitung, Herstellerdokumentation sowie weiterer schriftlicher Materialien zur Produktbeschreibung*“ vor. Nach BA (§ 6 (3)) ist das „*Maschinenprogramm mit Handbuch*“ sowie das „*Quellprogramm mit Dokumentation*“ zu übergeben. GL sieht unter 4.3 als Pflicht das Erstellen einer „*Benutzerdokumentation*“ vor. Das zusätzliche Erstellen einer „*systemtechnischen Dokumentation*“ ist optional, also ggf. gesondert zu vereinbaren.

In den BVB findet sich unter § 16 Nr. 1 die Pflicht, „*eine ausführliche Dokumentation (Programmentwicklungsdokumentation, Programmdokumentation) sowie sonstige programmbezogene schriftliche Unterlagen in deutscher Sprache*“ zur Verfügung zu stellen, wobei auf eine Vereinbarung im Erstellungsschein verwiesen wird.

⁵¹² Bei BA wird von der umfassenden Anpassung und Ergänzung einer Standardsoftware ausgegangen, weshalb diese, nebst „*umfassender Parametrisierung, Änderung und Erweiterung dieser Software*“ (§ 1 (1) b)) zu liefern ist.

SN-S spricht von der Übergabepflicht von „*Bedienungsanleitung (Bedienungshandbücher) und Installationsanweisung*“ (Nr. 1.2), während sich bei SN-B (1.3.2) die weitestgehende Formulierung findet:

„An Dokumentationen sind deshalb dem AG in verständlicher und für die jeweilige Adressatengruppe geeigneter Form zu erstellen und auszuhändigen:

- eine ausführliche Programmbeschreibung mit Kommentaren und Hinweisen auf die entsprechenden Passagen des Quellcodes,*
- Programmier-Anweisungen/Entwicklungsdokumentation, Datenmodell,*
- Installationsdokumentation,*
- Bedienungsanleitung.*

Die Frage der Quellcodeübergabe behandeln die Muster ebenfalls nicht einheitlich.

MA (§ 1), BA (§ 6 (3)), SN-B (5.2.) verlangen die Übergabe des Quellcode, bei KO (§ 2, 2.1), GL (9.1 (2) ist sie (nur) bei gesonderter Vereinbarung geschuldet, bei SN-S, HE und in den BVB ist sie nicht vorgesehen.

Über die Lieferung von Software und Dokumenten hinausgehende Pflichten finden sich als Option bei GL (6.3, 7.1) das Erstellen eines Pflichtenheftes und in den BVB (§ 1 Nr. 1 a)) grundsätzlich das Erstellen des „*DV-technischen Feinkonzeptes*“.

Die Vielfalt der verwendeten Begriffe hinsichtlich der Dokumentenlieferung trägt hier zur Verwirrung bei, zumal es an einer Definition oder der Erklärung des Zweckes der Übergabe fehlt. Einen ersten Ansatz zur Klärung enthält SN-B, wenn er die Geeignetheit für eine - allerdings nicht näher bezeichnete - Adressatengruppe verlangt. Es ist hier wohl davon auszugehen, dass Bedienungsanleitung, Benutzungsanleitung, Bedienerdokumentation für den Endanwender, d.h. den die Software (lediglich) benutzenden Personenkreis konzipiert werden sollen. Die Entwicklungsdokumentation, Entwicklerdokumentation oder systemtechnische Dokumentation ist demgegenüber für die Entwicklung, d.h. für den in der jeweiligen Programmierung sachverständigen Personenkreis vorgesehen. Angesichts der Bedeutung, die einer Dokumentation insbesondere für die Änderung beizumessen ist, erstaunt diese Ungenauigkeit. Nur bei GL, der insoweit im Erstellungsschein unter 1.3. auch nähere Angaben zur Qualität der

geforderten Dokumentation macht, klingt der eigentlich Zweck der Überlassung einer „systemtechnischen Dokumentation“ an:

*„Der Kunde erhält eine Benutzerdokumentation nach DIN 66285, die im übrigen folgende Anforderungen zu erfüllen hat: _____
Die Benutzerdokumentation wird dem Kunden in _____ facher Ausfertigung als Printversion und auf geeignetem Datenträger im _____-format zur Verfügung gestellt.
Der Kunde erhält eine systemtechnische Dokumentation, die es einem durchschnittlich erfahrenen Programmierer ermöglicht, den Programmplan und das Know-How nachvollziehen zu können, und die im übrigen folgende Anforderungen zu erfüllen hat: _____
Die Dokumentation wird dem Kunden in _____ facher Ausfertigung als Printversion und auf geeignetem Datenträger im _____-format zur Verfügung gestellt.“*

III. Spezifikationen

Die Muster gehen bei der Spezifikation der Leistung von unterschiedlichen Projektständen aus. Dabei wird nicht immer deutlich, in welcher Phase sich das Projekt tatsächlich befindet.

Während nach KO und augenscheinlich auch nach MA bei „Null“ begonnen wird, setzen GL und HE voraus, dass bereits erste Grundsteine für eine konkretere Leistungsbeschreibung gelegt wurden. Bei SN-S und den BVB liegt eine „fachliche Feinspezifikation“ bzw. ein „fachliches Feinkonzept“ vor. Bei SÖ findet sich als Anlage 2 eine „Funktionsbeschreibung“, die ohne Hinweis auf weitere Planungsleistungen oder -erfordernisse offenbar als ausreichende Grundlage für die vertragliche Vereinbarung angesehen wird. Bei BA und SN-B schließlich liegt zum Zeitplanpunkt des Vertragsschlusses ein Pflichtenheft vor.

Auf die Regelung auch im Hinblick auf die Verantwortung für die Spezifikation wird in der vorgenannten Reihenfolge, also nach dem Umfang erbrachter Vorarbeiten im Einzelnen eingegangen.

KO bietet die Optionen der Pflichtenhefterstellung durch den Kunden oder den Anbieter an, und formuliert dementsprechend (§ 1 Nr. 1.3

Pflichtenheft):

„(Variante: Pflichtenhefterstellung durch Kunden)

Der Kunde erarbeitet zur Vorbereitung der Programmerstellung mit dem Anbieter ein Pflichtenheft. Hierbei wird er vom Anbieter durch Beratung unterstützt.

(Variante: Pflichtenhefterstellung durch Anbieter)

(1) Der Anbieter erarbeitet in einem Pflichtenheft die Aufgabenstellung, die vom Anbieter durch Software-Erstellung anschließend zu lösen ist. Der Anbieter hat hierbei den Ist- und Soll-Zustand für die vorgesehene Anwendung zu beschreiben. Der Anbieter steht für die Richtigkeit und fachgerechte Durchführung der Erstellung dieser Vorgaben ein. Die Erstellung erfolgt in laufender Abstimmung mit dem Kunden.

(2) Dem Kunden steht es frei, aufgrund des vom Anbieter erstellten Pflichtenheftes den Erstellungsauftrag an einen Dritten zu vergeben. In diesem Fall ist eine Vergütung für die Pflichtenhefterstellung geschuldet, deren Höhe im Leistungsschein näher zu bezeichnen ist.

(3) Der Kunde nimmt das Pflichtenheft ab. Unrichtigkeiten oder Unvollständigkeiten des Pflichtenheftes sind vom Kunden bei oder nach Abgabe zu rügen, spätestens aber vor dem Zeitpunkt, zu dem der Anbieter mit der Programmerstellung auf der Grundlage des Pflichtenheftes beginnt. Rügt der Kunde zu einem späteren Zeitpunkt Mängel, die für ihn vor diesem Zeitpunkt des Beginns der Programmerstellung erkennbar waren, trägt der Kunde die Mehrkosten, die aus der nachträglichen Berücksichtigung dieser Rüge entstehen. Für nicht aus dem Pflichtenheft selbst erkennbare Mängel bleibt der Kunde berechtigt, eine Rüge ohne Mehrkosten für ihn bei Abnahme der erstellten Software zu erklären.“

KO geht von einem zweistufigen Verfahren aus, bei dem Programmerstellung und Pflichtenhefterstellung streng voneinander getrennt sind und die Erstellung auch von einem Dritten durchgeführt werden kann⁵¹³. Die erste Phase muss vom Softwarebesteller abgenommen werden, d.h. dieser muss bestätigen, das vom Softwarehaus erarbeitete Pflichtenheft entspreche seinen Vorstellungen, und ihn trifft eine Rügepflicht für Mängel.

Bei MA übernimmt der Softwarebesteller die Pflichtenhefterstellung, wobei bestimmte Vorgaben zu beachten sind (**§ 2 Softwarespezifikation und Pflichtenheft**):

„Die Software wird vom Hersteller entsprechend den im Pflichtenheft ausgearbeiteten Anforderungen hergestellt. Das Pflichtenheft wird vom Anwender unter angemessener Beratung durch den Hersteller erstellt. Es soll die in DIN 66231 (Programmentwicklungsdokumentation) aufgeführten

⁵¹³ Nach den Ergebnissen im Allgemeinen Teil ließe sich freilich mit guten Gründen vertreten, dass die Spezifikationserarbeitung einen integralen Teil der Softwareentwicklung darstellt. Auch ist nicht einsichtig, warum die Pflichtenhefterstellung nur dann vergütet werden soll, wenn die Programmerstellung durch einen Dritten erfolgt. Diese wichtige und umfangreiche Arbeitsleistung wird von jedem Anbieter zumindest kalkulatorisch in den Preis eingerechnet werden müssen.

Punkte soweit erforderlich detailliert festschreiben. Ab Fertigstellung des ausgearbeiteten Pflichtenhefts wird dieses als Anlage zum vorliegenden Vertrag geführt.“

Bei GL findet sich wie bei KO eine Option, wonach das Softwarehaus die Erstellung des Pflichtenhefts übernehmen kann. Vorausgesetzt wird dabei allerdings das Vorliegen einer „Aufgabenstellung“ des Softwarebestellers. Wie bei KO ist eine Abnahme und eine Rügepflicht vorgesehen (**Nr. 7. Erstellung des Pflichtenheftes**):

7.1 Erstellung

- (1) Falls dies im Erstellungsschein (ES 1.2) vorgesehen ist, erarbeitet der Anbieter ein Pflichtenheft. Grundlage ist die Aufgabenstellung des Kunden.*
- (2) Das Pflichtenheft beschreibt den Ist-Zustand der derzeitigen EDV-Lösung und den Soll-Zustand der zu entwickelnden EDV-Lösung, einschließlich des Soll-Zustandes nach etwa vereinbarten Zusatzleistungen. Näheres regelt der Erstellungsschein (ES 4.1). Der Anbieter gewährleistet Richtigkeit und Vollständigkeit des Pflichtenheftes.*
- (3) Der Kunde ist zur zumutbaren Mitarbeit verpflichtet.*
- (4) Falls und soweit bei der Ausarbeitung des Pflichtenheftes das Bedürfnis zu einer Anpassung der vertraglichen Regelungen, insbesondere des Zeitplans nach Erstellungsschein, Vergütung nach Erstellungsschein oder Funktionsprüfung nach Erstellungsschein erkennbar wird, entscheiden die Vertragspartner einvernehmlich über die Anpassungen des Erstellungsscheins.*
- (5) Kommt keine Einigung zustande, hat der Kunde das Recht, die Erstellung des Pflichtenheftes schriftlich für gescheitert zu erklären. Das Projekt wird in diesem Fall nicht durchgeführt; der Anbieter erhält für bereits geleistete Arbeit eine Zusatzvergütung nach Erstellungsschein (ES 5.5), soweit nicht eine aufwandsabhängige Vergütung geschuldet ist.*

7.2 Abnahme Geltung

- (1) Das Pflichtenheft ist nach Fertigstellung vom Kunden innerhalb von 30 Arbeitstagen abzunehmen. Der Kunde prüft das Pflichtenheft auf Richtigkeit und Vollständigkeit*
- (2) Das abgenommene Pflichtenheft wird Teil des Erstellungsscheines (ES 1.1.1). Falls und soweit Widersprüche zur Aufgabenstellung des Kunden bestehen, hat das Pflichtenheft Vorrang.*
- (3) Mit der Abnahme des Pflichtenheftes beginnt der Anbieter mit der Realisierung.*

7.3 Mängel des Pflichtenheftes

- (1) Fehler, die der Kunde aus dem Pflichtenheft erkennen kann, sind bei Abnahme zu rügen, spätestens bevor der Anbieter mit Aufwendungen für die Umsetzung des Pflichtenheftes beginnt. Werden erkennbare Fehler später gerügt, so trägt der Kunde die Mehrkosten, die aus der nachträglichen Berücksichtigung der Rüge entstehen.*

(2) Fehler, die der Kunde nicht aus dem Pflichtenheft erkennen kann, können geltend gemacht werden wie vom Anbieter zu vertretende Fehler bei der Umsetzung des Pflichtenheftes.

7.4 Vertragsfortführung nach Pflichtenheft

Dem Kunden steht es frei, die Umsetzung des Projektes auf der Grundlage des gelieferten Pflichtenheftes mit einem dritten Unternehmer fortzuführen.

7.5 Anpassung nach Beginn der Realisierung

Ergeben sich nach Realisierungsbeginn Änderungen hinsichtlich der Leistungsbeschreibung, so ist das Pflichtenheft durch den Anbieter entsprechend fortzuschreiben.

7.6 Vergütung

Der Anbieter erhält für die Erarbeitung eine Vergütung nach Erstellungsschein (ES 5.2) und für die Fortschreibung eine Zusatzvergütung nach Erstellungsschein (ES 5.5).“

Betrachtet man die Ausarbeitung der Aufgabenstellung als eigenen Schritt, so findet sich hier ein dreistufiges Verfahren. Zunächst definiert der Softwarebesteller seine Anforderungen, diese setzt das Softwarehaus zunächst in ein abzunehmendes Pflichtenheft und dann in ein, ebenfalls abzunehmendes, Programm um. Interessant ist die Ausstiegsoption des Softwarebestellers. Sie enthält einen deutlichen Hinweis auf die Fragilität des Planungsprozesses, die sich aus den Unwägbarkeiten bei Projektbeginn ergibt. Die Einschränkung der Mitarbeitspflicht des Softwarebestellers auf das Zumutbare ermöglicht es, eine übermäßige Belastung von dessen Ressourcen, insbesondere seinen Mitarbeitern, zu vermeiden und einem ggf. erheblichen Kompetenzgefälle hinsichtlich der Softwareentwicklung Rechnung zu tragen⁵¹⁴.

Auch HE geht von einer vorvertraglichen Tätigkeit der Vertragsparteien aus. Sein Vertragsmodell strebt die ausgeprägteste Form der Zusammenarbeit der Vertragsparteien an. Es orientiert sich sehr weitgehend an dem oben (AT) vorgestellten Spiralmodell der Softwareentwicklung, also einem (informationstechnischen) Verfahren der Softwareentwicklung. Sowohl die für den Juristen schwer fassbare Terminologie als auch die Verfahrensweise selbst tragen zur Einzigartigkeit des Ansatzes von HE bei.

⁵¹⁴ Ob die Vertragsparteien freilich nicht besser daran täten, konkrete Pflichten zu benennen, sei dahingestellt.

„§ 2 Ablaufbeschreibung des zugrunde gelegten Entwicklungsmodells

Das Entwicklungsziel soll durch eine zyklische Abfolge von vier Teilprozessen (Quadranten) erreicht werden. Durch die Erhöhung des Entwicklungsstandes mit jedem Zyklus ergibt sich das modellhafte Bild einer Spirale (Spiralmodell).

Der erste Quadrant dient zur Identifizierung der Anforderungen, Alternativen, Risiken und Einschränkungen.

Die Bewertung dieser Faktoren im zweiten Quadranten ergibt die Entscheidung zur Gestaltung des weiteren Vorgehens.

Die Realisierung der sich aus der Bewertung ergebenden Teilziele erfolgt im dritten Quadranten.

Jeder Zyklus schließt mit einem Planungsquadranten ab, in dem sich die Parteien auf den Zeit- und Ressourcenbedarf für den nächsten Zyklusdurchlauf einigen. Änderungen des übergeordneten Zusammenbauwerks sind dabei zu prüfen.

Die Teilergebnisse der Quadranten gelten nur dann als erfüllt, wenn eine Entscheidung nach den Kriterien des jeweiligen Einzelquadranten vorliegt.

Die Parteien bilden einen Lenkungsausschuss, dem E (Anm.: Softwarebesteller) vorsteht. (...).

*Teil 2 Bestimmungen für die Entwicklungsphasen**§ 3 Identifikationsphase*

Zu Beginn der Zusammenarbeit legen die Parteien ihre bisherigen Tätigkeiten im Themenbereich des Vertragsgegenstandes schriftlich nieder und quittieren die gegenseitige Übergabe.

In jedem Zyklus der Spirale werden zunächst Anforderungen, Alternativen und Einschränkungen identifiziert. Ziel der von S (Anm.: das Softwarehaus) zu erbringenden Planungs- und Beratungsleistungen im ersten Quadranten ist es also, in enger Zusammenarbeit mit E (Anm.: Softwarebesteller) die Leistungsfähigkeit, Funktionalität und Änderungsfähigkeit des Systems oder eines Teils davon zu bestimmen. Dabei sollen Alternativen, wie die Entscheidung zwischen den Lösungen A oder B, Wiederverwendung einer bereits vorher bestehenden eigenen Implementation oder der Kauf eines fremden Standardproduktes sowie die Barrieren für die Wahl der Alternativen aufgestellt werden.

Das Ergebnis des Identifikationsquadranten bildet die Aufgabenstellung für die weiteren Entwicklungsphasen. Diese wird durch den Lenkungsausschuss gemeinschaftlich getroffen und zyklisch konkretisiert.“

Der Schwerpunkt der Verantwortung liegt hier bei dem Softwarehaus, wobei die regelmäßige gemeinsame Entscheidungsnotwendigkeit den Softwarebesteller in sehr weitgehender Form einbezieht.

Nach der oben (S. 135) bereits dargestellten Nr. 1.1 SN-S wird der Softwarebesteller mit der Verantwortlichkeit für die fachliche Feinspezifikation belastet. Die Erstellung eines Pflichtenheftes ist nicht vorgesehen.

In den BVB wird bei der Spezifikationsfrage die Orientierung an einem EDV-technischen Phasenkonzept besonders deutlich. Zu den Erstellungsleistungen gehört nach § 1 Nr. 1 a) ausdrücklich

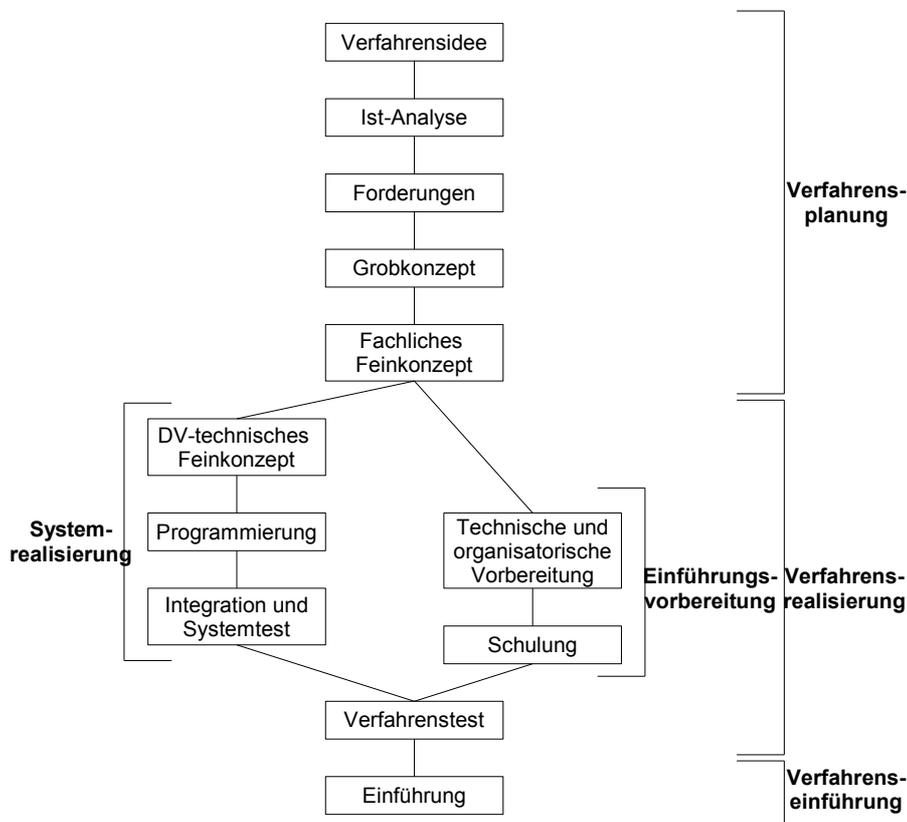
„das Erstellen des DV-technischen Feinkonzeptes“.

Unter § 1 Nr.2 wird dabei eine klare Abgrenzung zu Planungsleistungen versucht:

„Die Bedingungen gelten nicht für die Planung von DV-gestützten Verfahren.“

Im Phasenmodell gehören die Verfahrensidee, Ist-Analyse, Forderungen, das Grobkonzept und das fachliche Feinkonzept zur Verfahrensplanung, während u.a. das technische Feinkonzept, die Programmierung sowie das Testen der Verfahrensrealisierung und damit den BVB-Erstellung zugeordnet werden.

Übersicht über das Phasenkonzept



Nach den BVB liegt also bei Vertragsbeginn bezüglich der Softwarebestellung bereits ein idealerweise im Rahmen eines EDV-technischen Vorgehensmodelles entwickeltes fachliches Feinkonzept, d.h. eine klare Definition der Leistungsanforderung in fachlicher Hinsicht vor.

In der letzten Gruppe, SN-B und BA, bauen die Parteien auf einem bestehenden Pflichtenheft auf. Bei BA wurde dieses Pflichtenheft gemeinsam erstellt und vom Softwarehaus geprüft, d.h. die Verantwortung für die Geeignetheit des Pflichtenheftes für die nachgelagerten Projektschritte liegt alleine beim Softwarehaus.

„§ 4 Leistungsbeschreibung

(1) Das Projekt ist heute durch folgende Unterlagen beschrieben:

- a) ...*
- b) Gemeinsam erstelltes Pflichtenheft (Anlage Pflichtenheft)*
- c) ...*
- d) ...*

Der AN hat diese Unterlagen auf Vollständigkeit, Eindeutigkeit, Konsistenz und Realisierbarkeit geprüft.

Bei SN-B wird bereits in der Präambel deutlich, dass die verbindliche Definition der zu erbringenden Leistung ausschließlich Sache des Softwarehauses ist⁵¹⁵.

IV. Besondere Pflichten des Softwarehauses

KO, BA, GL, HE, SN-B und die BVB sehen besondere Pflichten des Softwarehauses vor. Mehrfach sind dabei Informations- und Berichtspflichten anzutreffen. So heißt es bei KO unter Nr. 1.12:

„(Hinweispflicht) Ist für den Auftragnehmer erkennbar, dass die Leistungsbeschreibung nach § 1.2 oder Anweisungen des Auftraggebers fehlerhaft, unvollständig, nicht eindeutig oder objektiv nicht ausführbar sind, muss er dem Auftraggeber diesen Umstand sowie die ihm erkennbaren Folgen hieraus unverzüglich schriftlich mitteilen. (...)

und unter 1.14:

⁵¹⁵ S.o., S. 132.

„(Unterrichtung) Der Auftragnehmer wird den Auftraggeber über den Fortgang der Erstellung entsprechend dem Zeitplan unterrichten und dem Auftraggeber auf Verlangen Einsicht in entsprechende Unterlagen und Auszüge hiervon gewähren sowie insbesondere ihm ein nach § 1.1 erste Variante erstelltes Feinkonzept vorlegen.“

GL formuliert ähnlich (**4.5 Leistungszeit**):

*„(1) ...
 (2) ...
 (3) Der Anbieter unterrichtet den Kunden über den Fortgang der Arbeiten. Sobald er Kenntnis von Hindernissen oder Verzögerungen erhält, welche die fristgemäße Leistungserbringung gefährden, teilt er dies dem Kunden unverzüglich mit.“*

Nach den BVB müssen u.a. Informationen zu Projekthindernissen weitere Ausführungen enthalten (**§ 3 Leistungen des Auftragnehmers**):

*„1. (...)
 2. (...)
 Der Auftragnehmer hat den Auftraggeber nach einem im Erstellungsschein festgelegten Zeitplan über den Stand der Arbeiten und die Einhaltung der Anforderungen an die Programme zu unterrichten und Zwischenergebnisse mitzuteilen. Darüber hinaus kann der Auftraggeber Einsicht in die entsprechenden Unterlagen und Auszüge hiervon verlangen. Einzelheiten und eine evtl. Vergütung für Zwischenberichte werden im Erstellungsschein vereinbart.
 Erkennt der Auftragnehmer, dass er die Ausführungsfristen nicht einhalten kann, hat er dem Auftraggeber unverzüglich die Gründe für die Verzögerung und ihre voraussichtliche Dauer mitzuteilen. Ein Anspruch auf Verlängerung der Ausführungsfristen besteht unbeschadet § 15 nicht.“*

Interessanterweise kommt nur in den BVB zum Ausdruck, dass die Auferlegung von Informationspflichten nicht unerhebliche Ressourcen verbrauchen kann, die u.U. eine Vergütung als angemessen erscheinen lassen.

BA konkretisiert die Berichtspflicht durch inhaltliche Vorgaben in **§ 14**

Information:

*„(1) Der AN erstellt für jeden Kalendermonat bis spätestens zum 7. Werktag des Folgemonats einen schriftlichen Bericht mit zumindest folgendem Inhalt:
 – Herstellungsgrad der Teilprojekte mit Angaben zur Einhaltung des Terminplanes und Verbrauch an Ressourcen.“*

- *Im Folgemonat geplante Arbeiten des AN und Beistellungen des AG; anstehende Entscheidungen.*
 - *Wesentliche Ergebnisse.*
 - *Überblick über das Projekt.*
- (2) ...“

Darüber hinaus sieht BA mindestens in 14tägigem Abstand stattfindende Projektbesprechungen vor, die vom Softwarehaus zu protokollieren sind
(§ 13 Besprechungen und Protokolle)

„(1) ...

(2) *Der AN erstellt über jede Projektbesprechung ein Protokoll, das die wesentlichen Erörterungspunkte und alle getroffenen Entscheidungen festhält und das er dem AG unverzüglich überlässt. Das Protokoll wird im Rahmen der Zuständigkeiten des jeweiligen Gremiums⁵¹⁶ verbindlich, wenn der AG nicht binnen fünf Werktagen schriftlich mit eigenem Formulierungsvorschlag widerspricht. Über den Widerspruch wird in der nächsten Projektbesprechung verhandelt.“*

Zur Sicherstellung der Mitarbeiterqualifikation finden sich bei BA, GL und in den BVB gesonderte Bestimmungen.

Bei GL werden entsprechende Erfordernisse kurz unter **5. Vergütung** im Erstellungsschein angedeutet:

„5.4.2 *Der Anbieter setzt Mitarbeiter mit folgender Qualifikation ein:*

_____“

Weitergehend ist in den BVB neben der Qualitätsanforderung eine, wenn auch nur ausnahmsweise zum Zuge kommende, Einflussnahmemöglichkeit auf Personalentscheidungen des Softwarebestellers vorgesehen (**§ 3 Leistungen des Auftragnehmers**):

„4. *Im Erstellungsschein kann die fachliche Qualifikation (z.B. Projektleiter, Systemprogrammierer) der bei dem Erstellen der Programme einzusetzenden Arbeitnehmer festgelegt werden. Wenn ein zur Vertragserfüllung eingesetzter Arbeitnehmer der Auftragnehmers durch einen anderen ersetzt werden muss, so geht dessen Einarbeitung zu Lasten des Auftragnehmers.*

⁵¹⁶ BA sieht in § 12 drei Gremien vor: Teilprojektteams, bestehend aus Teilprojektleitern von AN und AG; Projektleitung, bestehend aus allen (Teil-) Projektleitern, und Lenkungsausschuss, bestehend aus den Gesamtprojektleitern nebst Stellvertretern von AG und AN sowie einem weiteren Mitglied jedes Vertragspartners, das direkt der Geschäftsleitung unterstellt ist.

Der Auftraggeber kann mit schriftlicher Begründung den unverzüglichen Austausch eines Arbeitnehmers verlangen, wenn dieser wiederholt gegen vertragliche Pflichten verstoßen hat. Die durch den Wechsel entstehenden Kosten trägt der Auftragnehmer.“

Während in GL bereits aus der Verlagerung der Bestimmungen in den Erstellungsschein und dem geringen Umfang ein Hinweis auf die Einschätzung der Bedeutung zu entnehmen ist, ergibt sich dies in den BVB aus dem zur Wahl Stellen der entsprechenden Regelung.

Anders, nämlich bedeutend höher schätzt offenbar BA die Bedeutung der Personalfrage ein, wenn er unter **§ 15 Mitarbeiter** formuliert:

*„(1) Vorbehaltlich § 16 (Anm.: § 16 regelt Leistungserbringung durch Subunternehmer) erbringt der AN die Leistungen mit eigenen angestellten Mitarbeitern. Sie müssen die notwendigen Kenntnisse und Erfahrungen besitzen; der AG kann insofern Auskünfte verlangen. Die fachlichen Qualifikationen der Mitarbeiter des AN sind im Mitarbeiterplan (Anlage: Mitarbeiterplan) festgelegt.
(2) Die Projektleiter des AN (§ 11 Abs.2) müssen zumindest seit einem Jahr beim AN in einer vergleichbaren Position arbeiten.
(3) Die Vertragspartner werden einen Austausch von Projektmitarbeitern auf ein Mindestmaß beschränken und einander vorher konsultieren. Ein Projektleiter oder Mitarbeiter soll ausgetauscht werden, wenn der Vertragspartner hierfür nachvollziehbare Gründe vorträgt.“*

GL und BA sehen besondere Regeln für die Fehlerbeseitigung vor. Während GL dies unter **17 Haftung für Sachmängel** als eine Frage der Gewährleistung ansieht, verortet BA die entsprechende Regelung unter **§ 25 Fehlerbeseitigung** in den Abschnitt der Projektorganisation. BA behandelt sie als der Abnahme in § 26 nachgelagerte Bestimmung im Ergebnis inhaltlich ebenfalls als dem Bereich der Gewährleistung zugehörig.

GL:

*„17.5. Mängelbeseitigung/Rechtsfolgen
(1) Der Anbieter beseitigt Mängel, die im Abnahmeprotokoll festgehalten sind oder die vom Kunden innerhalb der Verjährungsfrist gemeldet werden.
(...) (Anm. Es folgen ausführliche Regeln zum einer Mängelmeldung folgenden Verfahren)“*

BA:

„(1) Der AG meldet Störungen und Fehler unverzüglich. Die Meldung kann zunächst mündlich erfolgen, ist jedoch spätestens am übernächsten

Werktag schriftlich zu wiederholen. Sie kann nur durch einen Projektleiter (§ 11 Abs.2) abgegeben werden oder durch eine andere Person, die die notwendigen Kenntnisse der Software und Qualifikation hat und dem AN als meldeberechtigt benannt wurde.

(2) Der AN unterstützt den AG nach den Regeln dieses Vertrages bis zur Aufklärung und Beseitigung des Fehlers. Soweit nicht ein Fehler der Leistung des AN vorliegt, sondern beispielsweise ein Bedienungsproblem oder ein Problem der Softwareumgebung, erhält der AN hierfür eine Vergütung nach der Kalkulation.

(3) Für die Fehlerbeseitigung gilt:

a) Bei einem Fehler der Klasse 1 (§ 21)⁵¹⁷ beginnt der AN sofort und mit allem Nachdruck mit den Maßnahmen, um zumindest das Problem in eine niedrigere Fehlerklasse zu verschieben, und setzt die Tätigkeit auch über die normale Arbeitszeit (werktags 8:00 bis 17:00 Uhr) mit aller zumutbaren Anstrengung fort. Spätestens nach drei vollen Werktagen darf kein Fehler der Klasse 1 mehr vorliegen.

b) Bei Fehlern der Klasse 2 beginnt der AN bei Fehlerkenntnis werktags bis 12:00 Uhr noch am selben Tag, bei späterer Fehlerkenntnis am nächsten Werktag vormittags mit der Fehlerbeseitigung und setzt sie solange fort, bis kein Fehler der Klasse 2 mehr vorliegt. Der AG kann eine Intensität wie bei Fehlern der Klasse 1 verlangen gegen Zahlung von Vergütungszuschlägen für die Leistung außerhalb der üblichen Arbeitszeit.

c) Fehler der Klasse 3 werden nach Zweckmäßigkeit und im Rahmen eines korrekten Konfigurationsmanagements alsbald oder später beseitigt. Es gilt die Fehlereinstufung nach § 21. Die Fehlerbeseitigung ist abzunehmen, z.B. durch ein Protokoll nach § 13 Abs.2. (Anm.: oben S. 147 wiedergegeben)

(4) Bei fremder Hard- oder Software kann der AN nur die Dienste des jeweiligen Lieferanten zur Verfügung stellen.“

Dass bei BA der Softwarebesteller (AG) und nicht das Softwarehaus (AN) originär die Fehlbeseitigung vornimmt (der AN „unterstützt“ bei der Fehlbeseitigung nach Abs. 2 lediglich), ist hier nicht ohne weiteres nachvollziehbar⁵¹⁸. Die Abgrenzung von (nicht vergütungspflichtiger) Gewährleistung und (vergütungspflichtiger) Softwarepflege weist keine deutlichen Konturen und im Zusammenhang mit der Gewährleistungsregelung in § 30 Abs. 5 eine erhebliche Komplexität auf⁵¹⁹.

Keines der Muster sieht eine Regelung für die Fehlerbehandlung im laufenden Projekt, d.h. insbesondere vor der Abnahme, vor. Dies ist aus juristischer Sicht verständlich, da ein „Fehler“ i.S. eines Mangels eine erbrachte Leistung voraussetzt. In der Softwaretechnik hingegen ist die

⁵¹⁷ BA unterscheidet in § 21 drei Klassen: Betriebsverhindernde Fehler (Klasse 1), betriebsbehindernde Fehler (Klasse 2) und Sonstige Fehler (Klasse 3).

⁵¹⁸ Möglicherweise handelt es sich auch lediglich um einen Druckfehler.

⁵¹⁹ In § 30 Abs.5 S. 2 BA heißt es zu der Gewährleistungsfrist: „Die Frist wird für Ansprüche aus Sachmängeln auch durch eine Fehlermeldung gehemmt, bis die Fehlerbeseitigung abgenommen ist (§ 25 Abs.3). Bei einem Fehler der Klasse 1 betrifft die Hemmung das ganze Teilprojekt. Im übrigen gilt § 203 BGB. Für jeweils 4 Fehler der Klasse 1 verlängert sich die Frist um drei Monate.“

Fehlerbeseitigung ein fortlaufender Prozess, der sich regelmäßig über die gesamte Lebensdauer des Projektes erstreckt.

Diesem Umstand wird nur bei HE, infolge seiner am – softwaretechnischen - Spiralmodell orientierten Ausrichtung, Rechnung getragen. Die Pflichten des Softwarehauses sind hier für die jeweiligen Projektphasen benannt:

„§ 3 Identifikationsphase“

(...) Ziel der von S (Anm.: dem Softwarehaus) zu erbringenden Planungs- und Beratungsleistungen im ersten Quadranten (Anm.: der erste Quadrant umfasst nach § 2 die Identifizierung der Anforderungen, Alternativen, Risiken und Einschränkungen) ist es also, in enger Zusammenarbeit mit E (Anm.: dem Softwarebesteller) die Leistungsfähigkeit, Funktionalität und Änderungsfähigkeit des Systems, oder eines Teils davon, zu bestimmen. (...)

§ 4 Bewertungsphase

Die identifizierten Alternativen werden im Bewertungsquadranten bewertet. (...) Die Entwicklungsteams werden von S geführt, falls der Lenkungsausschuss nicht eine abweichende Regelung festlegt. (...) Die Ergebnisse der Entwicklungsteams sollen aus folgenden Komponenten bestehen: Erstens aus einem inhaltlichen Modell, das die Anforderungen entsprechend ihrem Konkretisierungsgrad simuliert, zweitens aus einem Plan für das weitere Vorgehen, der den Bereich und den Grad der Umsetzung bestimmt, drittens aus dem konkreten Maßnahmenplan zur Implementierung im nächsten Quadranten. (...)

§ 5 Realisierungsphase

(...) S ist für die Umsetzung innerhalb der gemeinsam gebildeten Entwicklungsteams verantwortlich, falls der Lenkungsausschuss nicht eine abweichende Regelung definiert. Die Erstellungsleistung umfasst neben der softwaretechnischen Realisierung des Auftragmoduls eine Stellungnahme über die sinnvolle Durchführung des nächsten Spiralzyklus mit seinen Alternativen (...)

§ 6 Planungsphase

Die Planung des Entwicklungsprozesses für die nächsten Phasen erfolgt nach den gleichen Grundsätzen wie die der Produktentwicklung: Der Bestimmung der Entwicklungsziele und der Bewertung ihrer Alternativen folgt die Entscheidung über das Vorgehen im nächsten Spiraldurchlauf. Diese Leistungen werden unter der Leitung von S in enger Zusammenarbeit mit E erbracht.“

Die softwaretechnische Kompetenz des Softwarehauses bringt es nach HE also in eine leitende Rolle, die es aber nur bei entsprechender Mitarbeit des Softwarebestellers ausfüllen kann. Die Art der Formulierung entspricht weit mehr der Festlegung einer Methode als dem gewohnten juristischen Diktus. Den Vertragsparteien wird hier eine spezifizierte Leitlinie an die Hand gegeben, mit deren Hilfe die Vertragspartner ihr gemeinsames Projekt

abwickeln. Dieser Grundidee folgend werden eher Aufgabenbereiche zugewiesen, als Pflichten verteilt.

Bei GL findet sich als gesonderte Pflicht des Softwarehauses unter 10. optional das so genannte **Herstellen der Betriebsfertigkeit**.

„10.1 Herstellen der Betriebsfertigkeit durch den Anbieter

(1) Falls dies im Erstellungsschein (ES 1.1.2) vorgesehen ist, stellt der Anbieter die Betriebsfertigkeit des Systems sicher. Er erbringt hierbei insbesondere folgende Leistungen, soweit sie im Erstellungsschein vorgesehen sind:

- Installation des Lizenzproduktes*
- Integration des Lizenzproduktes*
- Migration von Datenbeständen*

Im Erstellungsschein können weitere Leistungen aufgeführt werden.

(2) Der Kunde wirkt hierbei in zumutbarer Weise mit. Die Vertragspartner klären frühzeitig, welche Beeinträchtigungen eines bereits genutzten Systems im Rahmen der Herstellung der Betriebsfertigkeit zu erwarten sind.

(3) Die Herstellung der Betriebsfertigkeit gilt als Teil der Hauptleistung und wird mit dieser gemeinsam gemäß VB 16. abgenommen. (...)“

Inwieweit es sich hierbei (v.a. was die Integration angeht) um eine Option handeln kann, ist angesichts des bei GL eingangs formulierten Vertragsziels von der Erarbeitung und Einführung einer auf den Kunden zugeschnittenen EDV-Lösung höchst fraglich. Die Lauffähigkeit zu erstellender Software in der technischen Umgebung des Bestellers ist ein wichtiger Planungsfaktor und dahingehende Überlegungen dürften in den allermeisten Fällen, d.h. bei vorhandener EDV, einen integralen Bestandteil der ersten Projektschritte bilden. Für die Problematik der Übernahme von Altdaten gilt prinzipiell dasselbe, wobei angesichts der Bedeutung dieser Frage erstaunt, dass sie nur bei GL auftaucht.

Eine Versicherungspflicht für Haftungsrisiken findet sich bei GL (Nr. 20.6) und in den BVB unter § 14 Nr.4, in beiden Fällen obligatorisch vorgesehen und nicht zur Option gestellt.

Im softwarebestellerfreundlichen Muster SN-B schließlich werden die Pflichten des Softwarehauses nicht unerheblich erweitert. So heißt es unter 2.4:

„Der AN hat, im Rahmen seines Pflichtenhefts und des Terminplans, die bei einem Projekt dieser Größenordnung und dieser Art üblichen Änderungen und Schwierigkeiten bereits berücksichtigt. (...)“

Weiterhin wird das Softwarehaus zur Anforderung verschiedener Leistungen beim Softwarebesteller verpflichtet. Dies gilt gem. 3.3 für Mitwirkungsleistungen:

„3.3 Abruf der Mitwirkung

Der rechtzeitige Abruf der Mitwirkungsleistungen des AG ist Sache des AN. Soweit bereits feste Termine für Mitwirkungsleistungen festgelegt sind (z.B. in Anlage 2), wird der AN auf solche Termine rechtzeitig hinweisen.“

Es gilt darüber hinaus auch für Testdaten:

„5.10.1 (...) Der AG wird rechtzeitig vom AN dahingehend informiert, welche Daten bzw. welche Unterlagen der AN wann benötigt. Für die Beschaffung der Testdaten ist der AG nur dann verantwortlich, wenn ihm die Daten hinreichend genau beschrieben worden sind. (...)“

Die Bestimmung unter 3.3 soll laut Anmerkung des Autors dazu dienen, eine Vertragsverletzung nach § 280 BGB als Ergebnis verspäteter Mitwirkung bzw. den Annahmeverzug zu vermeiden. Nicht zuletzt diese Intention setzt neben der auch in den anderen genannten Vorschriften zum Ausdruck kommenden einseitigen Risikoverlagerung das Muster erheblichen Zweifeln hinsichtlich der Eignung zum Einsatz in der Vertragspraxis aus, will der Verwender nicht sehenden Auges eine ggf. gerichtliche Korrektur in Kauf nehmen.

V. Besondere Pflichten des Softwarebestellers

In allen Mustern mit Ausnahme von HE wird vom Softwarebesteller die Bereitstellung von Ressourcen verlangt.

SÖ widmet den Mitwirkungspflichten eine eigene Anlage, die allerdings nur drei Paragraphen enthält.

„§ 1 Installation

Die Installation des Lizenzproduktes erfolgt durch den Lizenzgeber. Der Lizenznehmer verpflichtet sich, die nachfolgenden Mitwirkungspflichten zu erfüllen.

§ 2 Zugang

Dem Lizenzgeber bzw. dessen Beauftragten ist zum Zweck der Installation nach Vorankündigung der Zugang zu den Rechenanlagen des Lizenznehmers zu gestatten

§ 3 Leistungen des Lizenznehmers

- (1) Der Lizenzgeber erhält vom Lizenznehmer einen eigenen Raum zur Verfügung gestellt, soweit der Lizenzgeber sich für die Erbringung der Leistungen aus dem Lizenzvertrag beim Lizenznehmer aufhält. Der Lizenzgeber ist dabei berechtigt, das Telefon sowie sonstige Kommunikationseinrichtungen auf Kosten des Lizenznehmers in Anspruch zu nehmen, soweit dies für die Durchführung des Lizenzvertrages notwendig ist.*
- (2) Die Rechenanlagen des Lizenznehmers, auf denen das Lizenzprodukt installiert werden sollen, müssen in einem ordnungsgemäßen Zustand, insbesondere mit einem funktionierenden Betriebssystem ausgestattet sein. Der Lizenzgeber ist nicht verpflichtet, Störungen der Rechenanlagen erst zu beseitigen, um den Installationsvorgang vornehmen zu können.*
- (3) Der Lizenzgeber erhält vom Lizenznehmer die technischen Dokumentationen zu den bereits auf den Rechenanlagen des Lizenznehmers installierten Softwareprogrammen, soweit dies zur Installation des Lizenzproduktes notwendig ist.“*

Verwertbar ist hier der Grundgedanke, dass das Softwarehaus auf eine funktionsfähige EDV-Basis beim Softwarebesteller angewiesen ist. Bedarf allerdings bereits die Existenz eines funktionsfähigen Betriebssystems besonderer Vereinbarung, so dürften insgesamt erhebliche Zweifel an den Erfolgsaussichten eines umfangreicheren Softwareprojekts bestehen.

Bei KO ist dies unter „**§ 2 Lieferung und Installation**“ geregelt:

„2.2 (Mitwirkung des Kunden) Der Kunde stellt dem Anbieter zum Zweck der Programmentwicklung, Testdurchführung und Installation ausreichend Möglichkeiten der Anlagennutzung und Rechenzeiten zur Verfügung. Die benötigten Rechenzeiten sind im Software-Leistungsschein anzuführen und im Voraus terminlich festzuhalten. Zusätzliche Rechenzeit stellt der Kunde gegen angemessene Vergütung zur Verfügung.“

Durch die Vergütungspflicht für die außerplanmäßige Inanspruchnahme von Rechenzeit, also der Nutzung der EDV-Anlage des Softwarebestellers, wird betont, welche knappe Gut Rechenzeit darstellen kann. Zusätzlichen Bedarf, der zur Projektabwicklung ganz im Sinne des Softwarebestellers dient, mit einer Zahlungspflicht zu sanktionieren, mag auch dazu gedacht sein, die

Planungssorgfalt zu erhöhen. Es handelt sich gleichwohl um eine ungewöhnliche Regelung, die zudem gewissen Friktionen mit der allgemeinen Mitwirkungsregel in **§ 9 Pflichten des Kunden** ausgesetzt ist:

„9.2 (Unterstützungspflicht des Kunden) Der Kunde verpflichtet sich, die zur Herstellung des Werkes erforderlichen Tätigkeiten des Anbieters zu unterstützen. Insbesondere schafft der Kunde unentgeltlich alle Voraussetzungen im Bereich seiner Betriebsphäre, die zur Erstellung des Werkes erforderlich sind. Arbeitsräume müssen in den Geschäftszeiten zugänglich sein und erforderliche Arbeitsmittel zur Verfügung stehen. Testdaten und sonstige Informationen sind rechtzeitig beizustellen und mit dem Operating beauftragte Mitarbeiter rechtzeitig abzustellen.“

Durch die Sphärenabgrenzung wird deutlich gemacht, worauf es nach KO ankommt. Im Bereich des Softwarebestellers ist dieser selbst und alleine verantwortlich. Noch deutlicher akzentuiert findet sich dies bei BA, der unter **§ 5 Beistellungen und Abgrenzung** die Verantwortlichkeiten so aufzeigt:

„(1) Der AG stellt folgende Leistungen bei:

a) die Hardware (Anlage Hardware);

b) das Netzwerk (Anlage Netzwerk);

c) die System- und Datenbanksoftware (Anlage System- und Datenbanksoftware).

Der AG sorgt für Betriebsbereitschaft, Pflege und Wartung dieser Gegenstände.

(2) Außerdem stellt der AG sein besonderes versicherungstechnisches und versicherungsrechtliches Fachwissen zur Verfügung und wirkt nach § 17 mit.

(3) Die übrigen Leistungen, die notwendig sind, damit die Hardware (Abs.

1a) mit der neuen Software vertragsgemäß arbeitet, werden vom AN erbracht.“

Auch hier ist ausschließlich der Softwarebesteller für „seine“ EDV verantwortlich, mit der Einschränkung, dass die ordnungsgemäße Interaktion der neuen Software mit der vorhandenen EDV Sache des Softwarehauses wird (Abs. 3).

Die von BA vorgeschlagene weitere Grundregel in dessen Organisationsteil (**§ 17 Mitwirkung des AG**) ergänzt diese Anfangsbedingungen:

„(1) Der AG weiß, dass das Projekt nur realisiert werden kann, wenn er umfassend, fachkundig und rechtzeitig mitwirkt. Er wird den AN in dem Projekt unterstützen, womit nicht eine Übernahme von Verpflichtungen des AN gemeint ist. Er stellt zugunsten des Projektes seine Mitarbeiter in angemessenem Umfang von anderen Tätigkeiten frei. Er erbringt die Beistellungen (§ 5) rechtzeitig, gibt Informationen und gewährt dem AN im

notwendigen Umfang Zugang zu Räumen, Hard- und Software und Telekommunikation.

(2) Der AG schlägt rechtzeitig Testfälle und Testverfahren vor und übergibt Testdaten. Der AN fordert alle Mitwirkungsleistungen des AG rechtzeitig und spezifiziert an.“

Der Eingangssatz enthält hier einen möglicherweise für die Auslegung im Streitfall bedeutsamen Grundsatz, die nachfolgende Pflichtenbenennung bleibt in ihrem Umfang dabei allerdings begrenzt. Vor allem die Angemessenheit der Freistellung von Mitarbeitern dürfte einen regelmäßigen Anlass für kontrovers geführte Diskussionen bilden. Eine weitere Einschränkung erfährt die Position des Softwarehauses durch die Gestaltung der Mitwirkungsleistungen als „Holschuld“ nach Abs. 2 S. 2.

Bei MA ist die Mitwirkungspflicht nicht in vergleichbarer Weise relativiert:

„§ 6 Mitwirkungspflichten

- (1) Der Anwender ist im Rahmen des Zumutbaren zur angemessenen Mitwirkung bei der Programmherstellung verpflichtet. Die Mitwirkungspflicht umfasst insbesondere die Bereitstellung der für die Programmherstellung erforderlichen Informationen DV-technischer und projektorganisatorischer Art (Hardware und Betriebssysteme, eingesetzte Standardsoftware, Organisationspläne) sowie gegebenenfalls der Hardware, auf der das Programm später eingesetzt werden soll. (...)*
- (2) Sofern der Hersteller dem Anwender Entwürfe, Programmtestversionen oder ähnliches vorlegt, werden diese vom Anwender gewissenhaft geprüft. Reklamationen oder Änderungswünsche sind zu diesem Zeitpunkt anzumelden, soweit sie bereits erkennbar sind.*
- (3)(...)“*

Im Gegensatz zu BA formuliert MA eine Regel und nennt lediglich („insbesondere...“) Beispiele hierzu. Bemerkenswert ist auch die ausdrückliche Erwähnung der Prüfungspflicht des Softwarebestellers in Bezug auf Entwürfe, Programmtestversionen oder ähnlichem. Es wird dadurch deutlich, dass er sich selbst „gewissenhaft“ mit den Arbeitsergebnissen des Softwarehauses beschäftigen muss und bei entsprechendem Versäumnis mit Nachteilen, die allerdings nicht näher benannt sind, zu rechnen hat.

Selbst das bestellerfreundliche Muster SN-B wählt für die Pflichtenbenennung eine offene Formulierung, die jedoch mit einer Abrufverpflichtung kombiniert wird:

„3.2 Mitwirkung

Der AG ist zur Mitwirkung im Rahmen des erforderlichen Umfangs, insbesondere zu den nachgenannten Mitwirkungsleistungen verpflichtet:

3.2.1 Bereitstellung von Unterlagen, Testfällen, Testdaten

...

(Anm.: Diese und die folgenden Pünktchen finden sich im Original, der Text von 3.2. und 3.3 ist also vollständig wiedergegeben)

3.2.2 Bereitstellung von Räumen/Maschinen, Testsysteme

...

3.2.3 Bereitstellung von Personal

...

3.3. Abruf der Mitwirkung

Der rechtzeitige Abruf der Mitwirkungsleistungen des AG ist Sache des AN. Soweit bereits feste Termine für Mitwirkungsleistungen festgelegt sind (z.B. in Anlage 2), wird der AN auf solche Termine rechtzeitig hinweisen.“

Das Kontrastmuster, SN-S, geht darüber im Konkreten nicht wesentlich hinaus:

„6. Organisation des Projekts, Mitwirkung des Auftraggebers**6.1 Zusammenarbeit, Mitwirkung**

6.1.1 Das Projekt erfordert eine enge Zusammenarbeit der Parteien. Die Zeitplanung geht von einem erfolgreichen Ineinandergreifen der von den Parteien im Rahmen des Projektes zu erbringenden Leistungen und Vorleistungen aus. Die Parteien haben die bislang bekannten Pflichten beider Seiten im Aktivitäten- und Zeitplan (Ziffer 2.1) festgelegt.

6.1.2 Der Auftraggeber ist zur Mitwirkung auch im übrigen verpflichtet, insbesondere zur Beistellung von Testfällen, Testdaten, Testsystem und geeigneten Mitarbeitern (s.a. Ziffer 6.6.).

6.1.3 (...)“

GL regelt die Mitwirkung am ausführlichsten:

„14. Mitwirkung durch den Kunden**14.1. Allgemeine Mitwirkung**

Der Kunde unterstützt die Arbeiten des Anbieters in angemessener Weise. Weitergehende Regelungen bleiben unberührt.

14.2. Bereitstellung von Ressourcen

(1) Der Kunde stellt die in seiner Betriebssphäre liegenden Voraussetzungen sicher, die für die vertragsgemäße Erbringung der Leistungen des Anbieters erforderlich sind. Insbesondere stellt der Kunde dem Anbieter die im Erstellungsschein (ES 6.1) aufgeführten Räume, Mitarbeiter, Hardware und Software zu den dort vorgesehenen Konditionen und Zeiten zur Verfügung.

(2) Benötigt der Anbieter vom Kunden darüber hinausgehende Ressourcen, so sind diese angemessen zu vergüten. Näheres regelt der Erstellungsschein (ES 6.1).

14.3 Bereitstellung von Informationen

Der Kunde stellt dem Anbieter alle benötigten Informationen und Unterlagen zur Verfügung, insbesondere solche über Hardware, Software, Schnittstellen und Datenbestände, mit denen das Lizenzprodukt zusammenarbeiten soll; andernfalls benennt der Kunde taugliche Informationsquellen. Näheres regelt der Erstellungsschein (ES 6.2).

14.4. Bereitstellung von Testdaten

Der Kunde stellt dem Anbieter die im Erstellungsschein (ES 6.3) definierten Testdaten zur Verfügung.

14.5 Abruf von Mitwirkungshandlungen

(1) Der Anbieter ruft rechtzeitig und schriftlich konkrete Mitwirkungshandlungen ab.

(2) Der Anbieter bietet dem Kunden - soweit zumutbar - Unterstützung an. Er erhält für vom Kunden geforderte Unterstützungsleistungen eine Zusatzvergütung (ES 5.5).

14.6 Verletzung von Mitwirkungspflichten

(1) Erfüllt der Kunde in diesem Vertrag vorgesehene Mitwirkungspflichten nicht, nicht richtig oder nicht rechtzeitig, so rügt der Anbieter dies unverzüglich schriftlich.

(2) Der Kunde ist dem Anbieter zum Ersatz des aus der Verletzung der Mitwirkungspflicht entstehenden Schadens verpflichtet. Ein Zeitplan wird erforderlichenfalls angepasst.

(3) Der Anbieter kann dem Kunden eine angemessene Frist zur Erfüllung der Mitwirkungspflichten setzen. Nach fruchtlosem Ablauf der Frist kann der Anbieter vom Vertrag zurücktreten und neben Schadensersatz eine Vergütung verlangen, die den bisher erbrachten Leistungen entspricht. Der Kunde kann den Anbieter nach fruchtlosem Ablauf der Frist zu der Erklärung auffordern, ob er zurücktreten werde.“

GL benennt neben den Ressourcen hier „*Informationen*“, die vom Softwarebesteller zur Verfügung zu stellen sind. Diese Vereinbarung trägt dem Umstand Rechnung, dass das Softwarehaus in der Regel hinsichtlich der betrieblichen, d.h. v.a. fachlichen Gegebenheiten (wie branchenübliche Daten, Verfahren, Gebräuche o.ä.) auch bei eigener und sorgfältiger Ist-Analyse ein Informationsdefizit gegenüber dem Softwarebesteller hat. GL wählt hier die Regelbeispielmethode („*insbesondere*“).

Die Abrufpflicht für Mitwirkungshandlungen ist weiter formalisiert („*schriftlich konkrete*“) und die Folgen der Pflichtverletzung sind benannt, wobei hier eine unverzügliche schriftliche Rüge verlangt wird. Dem Softwarehaus wird also eine aktive Rolle zugewiesen, in der es die Mitwirkungshandlungen zunächst (rechtzeitig und schriftlich) abzurufen und bei Ausbleiben unverzüglich und ebenfalls schriftlich zu rügen hat.

Besondere Informationspflichten, die zum Teil mit Rügepflichten kombiniert werden, finden sich bei SN-S, GL und KO.

BA bezieht die Vorsorge bzgl. eines für besonders riskant angesehenen Bereiches in das Pflichtenprogramm des Softwarebesteller ein:

„§ 31 Haftung

(1) (...)

(2) *Der AG ist im Rahmen der Schadensminderungspflicht zur Datensicherung und zur Virenabwehr nach dem Stand der Technik verpflichtet.“*

Eine ähnliche Bestimmung findet sich auch bei SN-S:

„9. Haftungsbeschränkung“

(...)

9.9 Datensicherung

Der Auftraggeber ist für eine regelmäßige Sicherung seiner Daten verantwortlich. Bei einem vom Auftragnehmer verschuldeten Datenverlust haftet der Auftragnehmer deshalb ausschließlich für die Kosten der Vervielfältigung der Daten, der zu erstellenden Sicherheitskopien und für Kosten der Wiederherstellung der Daten, die auch bei einer ordnungsgemäß erfolgten Sicherung der Daten verloren gegangen wären.“

Dort werden dem Softwarebesteller zusätzlich weitergehende Nacherfüllungsmöglichkeiten eingeräumt:

„8.5 Minderung oder Rücktritt

(...)

Die Nacherfüllung gilt nicht schon mit dem zweiten Nacherfüllungsversuch als endgültig fehlgeschlagen. Vielmehr steht dem Auftragnehmer während der Nachfristen die Anzahl der Nacherfüllungsversuche in Abhängigkeit von der Art des Mangels, den besonderen Umständen (Personal u.ä.) sowie der Art der betroffenen Software (Beteiligung Dritter u.ä.) frei.“

In den BVB schließlich finden sich bedeutsame Einzelregelungen hinsichtlich des Treffens von Entscheidungen im Rahmen von Änderungen und Ergänzungen sowie der Beschreibung von Mängeln. Nach § 3 Nr. 1 Abs.2 S.2 hat der Softwarebesteller im Fall einer durch das Softwarehaus bemängelten Leistungsbeschreibung

„unverzüglich über eine Änderung der Leistungsbeschreibung (§ 5) oder seiner Forderungen zur Vertragsausführung (Nummer 3) zu entscheiden.“

Im Rahmen der Regelungen zur Gewährleistung in § 12 trifft den Softwarebesteller nach den BVB folgende Pflicht:

„4. Macht der Auftraggeber Mängel geltend, teilt er dem Auftragnehmer mit, wie sich die Mängel bemerkbar machen; dabei müssen die im Erstellungsschein festgelegten Unterlagen für die Mängelbeseitigung zur Einsichtnahme oder Anforderung zur Verfügung stehen. Benötigt der Auftragnehmer weitere Unterlagen, hat der Auftraggeber die Unterlagen unverzüglich zur Verfügung zu stellen. Darüber hinaus hat der Auftraggeber den Auftragnehmer bei der Mängelbeseitigung in dem im Erstellungsschein festgelegten Umfang zu unterstützen.“

Die beiden wichtigen hier enthaltenen Elemente der Mitwirkungspflicht beziehen sich auf den Umgang mit Problemen im Projektverlauf. Der Softwarebesteller ist dabei gefordert, diese nach Vermögen zu konkretisieren und nach eigenen Kräften bei der Abhilfe mitzuwirken, sowie - beispielhaft für mangelhafte Leistungsbeschreibung - unverzüglich erforderliche Entscheidungen zu treffen.

VI. Veränderungen (Change Management)

Das Change Management wird in allen Mustern mit Ausnahme von SÖ behandelt. Bei HE ist es Bestandteil des Entwicklungsprozesses und bedarf keiner gesonderten Regelung. Die Initiative soll dabei vom Softwarebesteller ausgehen, lediglich bei SN-S ist auch ein Anstoß für Änderungen vom Softwarehaus vorgesehen. Eine bestimmte Form der Änderungsmeldung wird nicht verlangt.

Eine prägnante Formulierung findet sich bei MA:

*„§ 5 Nachträgliche Änderungswünsche
(1) Änderungswünsche des Anwenders im Hinblick auf den Funktionsumfang, die Programmstruktur, die Bildschirmgestaltung oder sonstige Merkmale muss der Hersteller nicht berücksichtigen, soweit sie eine Abweichung vom ursprünglichen Vertragsinhalt darstellen, insbesondere nicht mit dem der Programmerstellung zugrunde gelegten Pflichtenheft oder sonstigen Leistungsbeschreibungen übereinstimmen.
(2) Dem Hersteller steht es frei, die gewünschten Änderungen gegen ein angemessenes zusätzliches Entgelt zu berücksichtigen. (...“*

Nach MA stellt das Abweichen vom Pflichtenheft eine vergütungspflichtige Ausnahmesituation dar. Basis für einen solchen Gedanken ist die Annahme, den Regelfall bilde ein im Pflichtenheft vollständig erfassbares Vertragsziel. Diese Annahme steht im Widerspruch zu den aus der Praxis der Softwareentwicklung gewonnenen Erkenntnissen, wie sie sich konsequent nur bei HE wieder finden.

Eine vergleichbar rudimentäre Regelung findet sich auch bei KO:

„1.6. (Leistungsänderungen) Alle nach Vertragsschluss erfolgenden Änderungen des Leistungsumfanges werden nur dann Vertragsinhalt, wenn sie vereinbart und im Software-Leistungsschein aufgenommen werden. Der Anbieter behält sich vor, die unter § 7 vereinbarte Vergütung bei vereinbarten Änderungen des Leistungsumfanges angemessen zu erhöhen. Der Auftragnehmer kann für eine erforderliche umfangreiche Prüfung, ob und zu welchen Bedingungen die gewünschte Änderung durchführbar ist, eine angemessene Vergütung verlangen, sofern er den Auftraggeber auf die Notwendigkeit der Prüfung hinweist und der Auftraggeber einen entsprechenden Prüfauftrag gibt (vergütungspflichtige Änderungsprüfung).“

Wie bei MA gibt es keinen Anspruch des Softwarebestellers auf Durchführung von Änderungen, erneut vor dem Hintergrund einer vollständigen Erfassung des Vertragszieles bei Vertragsabschluss. KO hält Änderungen allerdings nach dieser Formulierung insofern für regelungsbedürftig, als dem - änderungsbereiten - Softwarehaus durch die Prüfung eines Änderungsverlangens erheblicher Aufwand entstehen kann und damit eine Vergütung angemessen ist. Das Muster betont dabei die Notwendigkeit einer Vereinbarung, d.h. einer (kosten-) bewussten ausdrücklichen Entscheidung des Auftraggebers.

In den Mustern BA, GL, SN-B, SN-A und den BVB sind deutlich umfangreichere mehrstufige Verfahren für Änderungen vorgesehen. Die Änderungsregelung gehört bei GL, SN-B und SN-A sowie den BVB mit Abstand zu den umfangreichsten Regelungen des ganzen Vertrages. Bei BA, SN-B, GL und in den BVB kann der Softwarebesteller das Durchführen von Änderungen verlangen, wobei GL auch die Möglichkeit eines Ausschlusses des Änderungsrechts ausdrücklich vorsieht. Liegt das Änderungsverlangen vor, hat das Softwarehaus innerhalb einer vorgegebenen Zeit einen Vorschlag bzgl. des weiteren Vorgehens zu

unterbreiten. Alle Muster sind vom Bemühen gekennzeichnet, eine Kalkulierbarkeit der Änderungsfolgen zu erreichen und vor allem eine Verzögerung des Projektes durch Änderungen zu verhindern.

BA enthält die vergleichsweise kürzeste Vorschrift:

„§ 22 Leistungsänderungen

(1) Der AG kann jederzeit Leistungsänderungen und -ergänzungen verlangen, die im Rahmen des Projektes liegen, es sei denn, sie wären für den AN unzumutbar. Für kleine Änderungen (vorausgeschätzter Aufwand bis fünf Arbeitstage) genügt die Festlegung durch das Teilprojektteam nach § 13 Abs.2, wenn sich dadurch der Terminplan nicht ändert. Im Übrigen gilt § 12 Abs.3⁵²⁰.

(2) Der AN schätzt sofort ab, ob der Mehraufwand mehr als neun Arbeitstage beträgt. In diesem Fall verlangt er vom AG eine schriftliche Darstellung der Vorgaben und erstellt auf dieser Grundlage ein schriftliches Angebot. Hierbei ist er an die Kalkulation (Anlage Kalkulation) gebunden. Wenn der AG nichts anderes entscheidet, setzt der AN zunächst die Tätigkeiten nach den bisherigen Vorgaben fort.

(3) Falls sich die Vertragspartner nicht über die Vergütung und eine Änderung des Terminplans einigen, entscheidet in Bezug auf den Terminplan die Schlichtung (§ 23); in Bezug auf die Vergütung bleibt der Streit offen.

(4) Alle Vorgänge in diesem Zusammenhang sind aufs Äußerste zu beschleunigen, um eine Projektverzögerung zu vermeiden.“

Das Softwarehaus muss einem Änderungsverlangen in zwei Fällen nicht entsprechen. Einmal, wenn es nicht „im Rahmen des Projektes“ liegt und zum anderen wenn es „unzumutbar“ ist. Es werden kleine Änderungen (bis 5 Arbeitstage Aufwand), Änderungen mit 5 bis 9 Arbeitstagen Aufwand und Änderungen mit mehr als 9 Tagen Aufwand unterschieden. Je nach Umfang werden unterschiedliche Verfahren für das weitere Vorgehen bestimmt, wobei für mittleren Aufwand eine Regelungslücke besteht. Konkrete Fristen sind nicht vorgesehen, BA beschränkt sich insoweit auf die Generalklausel des Abs. 4. Bei den umfangreichsten Änderungen findet ein formalisierter Einigungsversuch auf der bereits verhandelten Vergütungsbasis und anhand einer Änderungsspezifikation des Softwarebestellers und darauf beruhender Kalkulation des Softwarehauses statt. BA versucht also, Änderungen in möglichst enger Anlehnung an bereits bestehende Regelungen abzufedern

⁵²⁰ In § 12 Abs. 3 findet sich folgende Regelung: *„Ein Teilprojektteam kann sein Budget unter dem Vorbehalt der Zustimmung der Projektleitung um höchstens 20 % erhöhen. Die Projektleitung kann den Spielraum nach § 7 Abs.2 (bei der Vergütung) und § 19 (beim Terminplan) ausschöpfen. Weitere Änderungen bedürfen der Zustimmung des Lenkungsausschusses.“*

und die Vertragspartner auf möglichst konkrete Änderungsinhalte und -folgen zu verpflichten.

Weiter ausgeformt ist das ansonsten dreistufige Verfahren bei GL:

„12. Change Management

12.1 Änderungsrecht

- (1) *Soweit dies nicht im Erstellungsschein (ES 1.5) ausgeschlossen ist, kann der Kunde ab dem Beginn der Realisierung bis zur Abnahme schriftlich verlangen, dass die gemäß VB 4.1 im Erstellungsschein festgelegte Leistungsbeschreibung geändert wird (Änderungsverlangen).*
- (2) *Der Anbieter kann Änderungen aufgrund seiner anderweitigen betrieblichen Auslastung, seiner mangelnden Leistungsfähigkeit oder aufgrund sonstiger Gründe als unzumutbar schriftlich ablehnen.*
- (3) *Der Anbieter prüft unverzüglich das Änderungsverlangen. Er gibt dem Kunden einen schriftlichen Hinweis, wenn das Änderungsverlangen erkennbar fehlerhaft, unvollständig, unklar oder unmöglich ist; er weist auf möglicherweise drohende Mängel hin. Der Kunde entscheidet unverzüglich über ein Änderung des Änderungsverlangens. VB 17.3 (1) bleibt unberührt (Anm.: diese Regelung schließt eine Gewährleistungspflicht für den Fall aus, dass der Softwarebesteller auf die Änderung trotz geäußerter Bedenken des Softwarehauses besteht)*

12.2 Ausführliche Prüfung

Falls eine umfangreiche Prüfung notwendig ist, ob und zu welchen Bedingungen eine Änderung durchgeführt werden kann, gibt der Anbieter dem Kunden einen entsprechenden Hinweis. Wenn der Kunde dem Anbieter einen Prüfauftrag erteilt, hat der Anbieter das Recht, für diesen zusätzlichen Aufwand eine Zusatzvergütung nach dem Erstellungsschein abzurechnen (ES 5.5). Die Vertragspartner legen einvernehmlich die Frist fest, innerhalb derer die Ergebnisse der Prüfung schriftlich vorgelegt werden.

12.3 Anpassung von Vergütung, Zeitplan, u.a.

- (1) *Falls und soweit die Umsetzung eines Änderungsverlangens eine Anpassung der vertraglichen Regelungen insbesondere des Zeitplanes, der Vergütung oder der Funktionsprüfung nötig werden lässt, hat der Anbieter ein entsprechendes Anpassungsverlangen mitzuteilen.*
- (2) *Falls das Anpassungsverlangen nicht innerhalb von 30 Arbeitstagen nach Zugang des endgültigen Änderungsverlangens, ggf. innerhalb der nach VB 12.2 Satz 3 festgelegten Frist erfolgt, ist eine spätere Anpassung wegen des Änderungsverlangens ausgeschlossen.*
- (3) *Die Parteien einigen sich innerhalb von 10 Arbeitstagen nach Zugang eines Anpassungsverlangens über die Änderung der Leistungsbeschreibung und die sonstige Anpassung des Erstellungsscheines.*

12.4 Arbeitsstopp

Führt der Anbieter bei Zugang des Änderungsverlangens Arbeiten aus, die bei Umsetzung des Änderungsverlangens nicht mehr verwendbar wären, kann der Kunde verlangen, dass der Anbieter diese bis zur Anpassung der Leistungsbeschreibung bzw. des Pflichtenheftes unterbricht; gleiches gilt für neu anstehende Arbeiten. Sobald der Anbieter erkennt, dass derzeitige oder anstehende Arbeiten bei der Umsetzung des Änderungsverlangens nicht mehr verwendbar wären, hat er den Kunden auf diese Möglichkeit unverzüglich hinzuweisen.

12.5 Verbindlichkeit

- (1) Die Änderung ist verbindlich, sofern der Anbieter nicht innerhalb von 30 Arbeitstagen ab Zugang des Änderungsverlangens als unzumutbar ablehnt, eine Prüfung nach VB 12.2 oder Anpassungen des Erstellungsscheines nach VB 12.3 fordert.
- (2) Falls ein Anpassungsverlangen nach VB 12.3 vorliegt, ist die verlangte Änderung nur dann verbindlich, wenn über das Anpassungsverlangen fristgerecht eine Einigung erzielt wurde (VB 12.3 (3)).
- (3) Wird eine Änderung verbindlich, so ist sie in die Leistungsbeschreibung einzuarbeiten, die weiteren Anpassungen gemäß VB 12.3 sind in den Vertrag umzusetzen. Ein Zeitplan (ES 3.) wird des weiteren um den Zeitverlust angepasst, der durch die Durchführung des Änderungsverfahrens verursacht wurde. Soweit die Ressourcen des Anbieters nicht anderweitig eingesetzt werden konnten, ist dieser zur Geltendmachung einer Zusatzvergütung nach dem Erstellungsschein berechtigt (ES 5.5).
- 12.6 Scheitern des Änderungsverlangens
Wird ein Änderungsverlangen nicht verbindlich, so sind die Arbeiten auf der Grundlage der bisherigen Bestimmungen weiterzuführen. VB 22.2 bleibt unberührt.“ (Anm.: Dort wird dem Softwarebesteller ein Kündigungsrecht gem. § 649 BGB eingeräumt)

Auch bei GL führt ein Änderungsverlangen des Softwarebestellers zunächst zu einer Prüfungspflicht, die in einen weitergehenden und vergütungspflichtigen Prüfungsauftrag münden kann. Das Softwarehaus hat innerhalb von 30 Arbeitstagen zu prüfen, ob das Änderungsverlangen mangelhaft ist, ob es Anpassungen von Zeitplan oder Vergütung erfordert und welche Auswirkungen es auf derzeitige und anstehende Arbeiten hat. Die Vertragspartner haben, nachdem die Folgen des Änderungsverlangens vom Softwarehaus als vertragsändernd eingestuft wurden, 10 Arbeitstage Zeit, um v.a. Fristen und Termine neu festzulegen. Gelingt dies nicht, gilt der bestehende Vertrag. Auch hier kann der Softwarebesteller eine Änderung nur eingeschränkt erzwingen, wobei das Softwarehaus zu fristgerechter Reaktion angehalten werden soll.

Eine straffere Fristenregelung in einem ähnlich formalisierten Verfahren findet sich bei SN-B, der dabei zwischen kleineren Änderungen, die das Softwarehaus „bestmöglich“ zu berücksichtigen hat, und größeren Änderungen, die ein förmliches Änderungsverlangen des Softwarebestellers erfordern, unterscheidet.

„2.4 Änderungen

Der AN hat, im Rahmen seines Pflichtenhefts und des Terminplans, die bei einem Projekt dieser Größenordnung und dieser Art üblichen Änderungen

und Schwierigkeiten bereits berücksichtigt. Infolgedessen ist der AG berechtigt, die sich im Rahmen einer Konkretisierung entstehenden Ergänzungs- und Änderungswünsche anzubringen, die der Auftragnehmer bestmöglich berücksichtigt wird.

2.4.1 Der AG ist aber berechtigt, größere Änderungen zu verlangen, wenn sich solche aus der Entwicklung seines Betriebs und seiner betrieblichen Strategie ergeben. In diesem Falle wird der AG ein förmliches Änderungsverlangen an den AN stellen, wonach das Pflichtenheft entsprechend anzupassen und zu ergänzen ist.

Der AN wird dieses Änderungsverlangen kurzfristig spätestens innerhalb einer Woche beantworten und dabei auch mitteilen, ob er dieses Änderungsverlangen näher prüfen muss. Er wird dabei insbesondere angeben, welche Änderungen sich aus seiner Sicht gegenüber der Projektschätzung (siehe 1.2) ergeben, welche Auswirkungen auf die Termine dies womöglich hat und welchen Zeitaufwand er benötigt, um ein detailliertes Angebot in preislicher und terminlicher Hinsicht zur Ausführung des Änderungswunsches anzubieten.

Erkennt der AN bereits in diesem Stadium, dass die Ausführung des Änderungswunsches für ihn unzumutbar ist bzw. seine Kräfte übersteigt, so wird er auch dies bereits jetzt mitteilen.

2.4.2 Unterbreitet der AN ein Angebot innerhalb einer angemessenen Frist, so kann der AG entscheiden, ob er dieses Angebot annehmen will oder nicht. Er hat dies binnen einer Frist von zehn Kalendertagen wahrzunehmen. Erfolgt innerhalb dieser Frist keine Antwort des AG, gilt das Angebot als abgelehnt.

2.4.3 Nimmt der AG das Angebot an, wird die Änderung Bestandteil dieser Vereinbarung und der Terminplan entsprechend ergänzt. Die Änderung wird auch bei allen übrigen Unterlagen berücksichtigt.

2.4.4 Für die Prüfung des Änderungswunsches kann der AN, wenn er dies bei seiner Antwort binnen einer Woche entsprechend angekündigt hat, und für die Ausarbeitung entsprechender Unterlagen, insbesondere Ergänzungs-, Pflichtenhefte, eine Vergütung verlangen. Diese Vergütung erfolgt grundsätzlich nach Zeitaufwand, wenn nichts anderes zwischen den Parteien vereinbart wird. Die Berechnung erfolgt auf der Basis eines Stundensatzes gemäß dem in der Anlage 3 beigefügten Preisblatt. Der AN wird entsprechende Zeitnachweise führen und sich diese als Regiearbeiten vom AG zeitnah abzeichnen lassen. Die gleichzeitige Vorlage abgezeichneter Regiezettel / Zeitnachweise ist Voraussetzung für einen Zahlungsanspruch und nicht nur für die Fälligkeit.

2.4.5 Zeigt sich während der Bearbeitung des Änderungswunsches, gleich in welchem Stadium, dass es sich tatsächlich nicht um einen Änderungswunsch, sondern um einen Fehler des Pflichtenheftes oder der sonstigen Ausarbeitung seitens des AN handelt, werden die Vereinbarungen hinsichtlich der Vergütungspflichtigkeit und der Verschiebung von Terminen hinfällig. Die Vertragspartner nehmen je nach Erkennbarkeit dieses Fehlers eine angemessene Anpassung des Gesamtvertrages und dabei der Vergütung (in Analogie zur Verzugsregelung) vor.

2.4.6 Hinsichtlich etwaiger Minderungen und hinsichtlich des Mehraufwands im Zusammenhang mit Angeboten wird der AN jeweils einen Abgleich mit der Projektschätzung (siehe 1.2) vornehmen und diese dann, wenn

die Änderung Vertragsbestandteil wird, fortschreiben. Kommt keine Einigung über die Änderung zustande, obwohl der AN rechtzeitig ein Angebot unterbreitet hat, ist der AN nicht berechtigt, außerordentlich zu kündigen. Er hat lediglich das Recht – nach wie vor – nach § 649 BGB vorzugehen, also eine ordentliche Kündigung auszusprechen.“

Beruhet eine Änderung auf einem Fehler aus der Sphäre des Softwarehauses, insbesondere im Pflichtenheft, so ändert dies nach diesem Muster nichts an Vergütung und Terminen. Dies ist rechtlich durchaus nachvollziehbar, ändert allerdings nichts an dem faktisch möglicherweise erheblichen Mehraufwand. Dies Projekt kann dadurch in eine Krise geraten, für die hier keine Lösung vorgesehen ist.

Im softwarehausfreundlichen Parallelentwurf SN-S fällt die Regelung etwas kürzer aus.

„3. Änderungen der Leistungen

3.1 Änderungsverlangen

Beide Vertragspartner sind berechtigt, unter Angabe wichtiger Gründe den anderen Vertragspartner aufzufordern, über Änderungen dieses Vertrages oder der (einzelnen) fachlichen Feinspezifikationen zu beraten und zu verhandeln.

3.2 Prüfung des Änderungsverlangens

Soweit der Auftraggeber Änderungen in bereits verabschiedeten Spezifikationen wünscht, wird der Auftragnehmer, gegen Vergütung auf Zeit- und Materialbasis entsprechend der allgemeinen Preisliste des Auftragnehmers, den dabei entstehenden Aufwand prüfen, sowie ob die gewünschte Änderung durchführbar ist, und den Auftraggeber dann möglichst kurzfristig darüber informieren, welche Änderungen sich insbesondere hinsichtlich der Kosten und des Zeitplans voraussichtlich ergeben.

Soweit möglich und notwendig, wird der Auftragnehmer auch prüfen, inwieweit eine solche Änderung Auswirkungen auf bisher realisierte Leistungen und deren Nutzbarkeit hat.

3.3 Unterbrechung zur Prüfung von Änderungen

Gegen Vergütung der Ausfallzeiten kann der Auftraggeber bis zur Einigung über ein Änderungsverlangen teilweise oder vollständige Unterbrechung der Realisierung fordern. Eventuell vereinbarte Leistungsfristen und Zeitpläne verlängern sich dementsprechend um die Ausfallzeit sowie um die Zeit, die der Auftragnehmer benötigt, um nach seiner Unterbrechung die Wiederaufnahme der Arbeiten zu organisieren und die notwendigen Ressourcen wieder zur Verfügung zu stellen.

3.4 Änderungsfeinspezifikation

Die Parteien werden die gewünschten Änderungen in einer Änderungsfeinspezifikation schriftlich festlegen und gemeinsam verabschieden.

3.5 Mangel der Einigung über Änderung

Wird über ein Änderungsverlangen keine Einigung erzielt, werden die Parteien, soweit sie keine andere Vereinbarung treffen, das Projekt entsprechend der verabschiedeten Feinspezifikation (1.1.) realisieren.“

Zeitvorgaben werden hier nicht gemacht und das Vergütungsinteresse des Softwarehauses wird besonders betont. Es fehlt an einem mehrstufigen Verfahren, wie in den vorstehenden Mustern. Hervorgehoben wird zudem die beidseitige Verantwortlichkeit bei der Initiative zur Änderung (3.1) und bei der Ausarbeitung der Änderungsspezifikation (3.4). Die Prüfung von Änderungen darauf, ob sie Auswirkungen auf den Projektstatus haben, (nur) für den Fall vorzusehen, dass dies „möglich und notwendig“ ist, stellt eine durchaus zweifelhafte Option für das Softwarehaus dar. Sieht es von einer solchen Prüfung ab und stellen sich im Nachhinein Schwierigkeiten bei der Projektabwicklung heraus, dürfte ein Berufen auf diese Klausel einer (schieds-) gerichtlichen Kontrolle kaum standhalten.

In den BVB findet sich ein in weiten Teilen den vorstehenden Mustern entsprechendes Verfahren.

„§ 5 Änderung der Leistung

1. Der Auftraggeber kann bis zur Abnahme oder dem im Erstellungsschein vereinbarten Zeitpunkt schriftlich die Änderung der im Erstellungsschein festgelegten Anforderungen an die Programme verlangen. Der Auftragnehmer hat die geänderten Leistungen auszuführen, soweit sie ihm im Rahmen seiner betrieblichen Leistungsfähigkeit nicht unzumutbar sind. Sofern der Auftragnehmer nicht innerhalb von 21 Kalendertagen ab Zugang des Änderungsverlangens die Änderung als unzumutbar ablehnt oder ein Prüfung nach Absatz 2 geltend macht, hat der Auftragnehmer die Änderung durchzuführen.

Erfordert das Änderungsverlangen vom Auftragnehmer eine umfangreiche Prüfung, ob und zu welchen Bedingungen die Änderung durchführbar ist, so kann er hierfür eine Vergütung insoweit verlangen, als er den Auftraggeber schriftlich darauf hingewiesen und der Auftraggeber daraufhin den Prüfungsauftrag schriftlich erteilt hat; die Frist, bis zu deren Ablauf dem Auftraggeber das Ergebnis der Prüfung schriftlich mitgeteilt sein muss, ist einvernehmlich festzulegen.

2. Beeinflusst die Änderung einer Leistung oder eine Forderung zur Vertragsausführung vertragliche Regelungen, z.B. Preis, Ausführungsfristen, Abnahme, wird unverzüglich die durch die Änderung bedingte Anpassung im Erstellungsschein unter Berücksichtigung entstehender Mehr- oder Minderaufwendungen vereinbart. Sofern der Auftragnehmer die Anpassung des Erstellungsscheines nicht innerhalb von 21 Kalendertagen nach Zugang des Änderungsverlangens oder bis zum Ablauf der Frist gemäß Nummer. 1 Abs. 2 S. 2 schriftlich geltend macht, ist die geänderte Leistung im Rahmen der bestehenden Vereinbarungen auszuführen.

Der Auftragnehmer kann verlangen, dass die von der Leistungsänderung betroffenen Arbeiten bis zur Anpassung des Erstellungsscheines entsprechend der geänderten Leistung und Gegenleistung unterbrochen werden. Wird die Ausführung nicht vom Auftraggeber unterbrochen und

erkennt der Auftragnehmer, dass die zwischen Zugang des Änderungsverlangens und Anpassung des Erstellungsscheines auszuführenden Arbeiten im Falle der Durchführung der Änderung nicht verwendbar sind, hat er dies dem Auftraggeber unverzüglich mitzuteilen.

3. Kommt eine Anpassung des Erstellungsscheines nicht innerhalb von 21 Kalendertagen nach Zugang des Verlangens des Auftragnehmers zur Anpassung der vertraglichen Regelungen zustande, so werden die Arbeiten auf der Grundlage der bestehenden Vereinbarungen weitergeführt, soweit der Auftraggeber den Vertrag nicht gemäß § 649 BGB kündigt. Die Ausführungsfristen verlängern sich um die Zahl der Kalendertage, an denen in Folge des Änderungsverlangens bzw. der Prüfung des Änderungsverlangens gemäß Nummer 1 Abs. 2 die Ausführung unterbrochen wurde.

Der Auftragnehmer kann für die Dauer der Unterbrechung die vereinbarte Vergütung sowie die entsprechende Erhöhung einer vereinbarten Obergrenze bzw. die entsprechende Erhöhung eines vereinbarten Festpreises verlangen, wenn und soweit die von der Unterbrechung betroffenen Arbeitnehmer nicht anderweitig eingesetzt werden konnten und dem Auftraggeber dies schriftlich mitgeteilt wurde.“

Das formalisierte Schema Änderungsverlangen - Prüfung oder ausführliche vergütungspflichtige Prüfung durch das Softwarehaus und ggf.

Unterbrechung der Arbeiten - Entscheidung des Softwarebestellers ist auch hier vorgesehen. Die gegenseitigen Handlungsanforderungen und die Vergütungspflicht sind weitgehend ausgeglichen geregelt und erkennbar von den Gedanken getragen, Überraschungen für beide Vertragspartner zu vermeiden und die Behinderung des Projektfortganges zu minimieren. Dass dieser gewünschte Projektfortgang aber ohne Einvernehmen nicht zustande kommen kann, wird durch Abs. 2 S. 1 anschaulich, wenn hier von unverzüglicher Vereinbarung der Anpassung die Rede ist. Der Fortgang des Projektes kann bei Änderungen nicht einseitig erzwungen werden, er muss von beiden Vertragsparteien gewollt sein.

VII. Zusammenhang zwischen Leistungsbestimmung und Einstandspflicht

Hinsichtlich des Maßstabes für Leistungserbringung findet sich bei KO (§ 4 Abs.1), GL (Nr. 4.4 und Nr. 17 Abs.2⁵²¹), SN-S (Nr. 8.1.1), SÖ (§ 11 Abs.1) und HE (§ 13) eine Anlehnung an die gesetzliche Regelung des § 434 BGB.

⁵²¹ Die gesetzliche Regelung wird im Verhältnis zur Leistungsbeschreibung im Erstellungsschein ausdrücklich für subsidiär erklärt

Bei GL findet sich darüber hinaus unter „**8.1 Programmierarbeiten**“ die Anforderung eines Standards:

„Soweit nichts anderes vereinbart ist, wird das Programm nach dem aktuellen Stand des Software-Engineering und den Grundsätzen ordnungsgemäßer Berufsausübung erbracht.“

Dies ist insofern interessant, als nach den informationstechnischen Methoden der Softwareentwicklung bereits seit geraumer Zeit eine vollständige Definition des Leistungsgegenstandes, wie sie bei GL im Erstellungsschein vorgesehen ist, nicht vorgenommen wird.

SÖ verwendet einen ähnlichen Maßstab, wenn er in **§ 11 Abs.1** ausführt:

*„(1) Gewährleistungsumfang
Der Lizenzgeber steht dafür ein, dass das Lizenzprodukt die Hauptfunktionen im wesentlichen erfüllt und den anerkannten Regeln der Technik entspricht sowie nicht mit Fehlern behaftet ist, die den Wert oder die Tauglichkeit zu dem gewöhnlichen oder nach dem Vertrag vorausgesetzten Gebrauch aufheben oder mindern.“*

Dies entspricht dem Wortlaut des überholten § 459 Abs.1 S.1 BGB und lässt daher auch eine Hierarchie bzgl. der vertraglich vorausgesetzten und der gewöhnlichen Verwendung, wie sie sich jetzt in § 434 Abs.1 S.2 BGB findet, vermissen.

Die ausführlichsten Anforderungen finden sich bei BA unter „**§ 6 Abstrakte Leistungsvorgaben**“

„(1) Die Leistungen (auch die Beistellungen des AG nach § 5) haben folgenden abstrakten Leistungsvorgaben zu genügen:

- Stand der Technik bei Vertragsabschluss;*
- Verfahrensnormen nach DIN/ISO 9000ff. oder andere in Hinblick auf die Qualitätssicherung gleichwertige Verfahrensregeln;*
- Grundsätze ordnungsmäßiger Datenverarbeitung;*
- Vorgaben der Bundesanstalt für Finanzdienstleistungsaufsicht.*

(2) Individualsoftware ist so zu erstellen und zu dokumentieren, dass sie die Anforderungen an Zeitverhalten, Ergonomie, Fehlertoleranz, Wartbarkeit (auch durch Dritte) und Interoperabilität erfüllt und durch möglichst einfache Schnittstellen mit der angrenzenden Standardsoftware verbunden ist (Sicherung der Kompatibilität zu künftigen Ständen der Standardsoftware).“

Dieser Anforderungsliste ist allerdings eher der Wunsch zu entnehmen, geltende Standards möglichst weitgehend zu erfassen, als dass sie eine praktisch verwertbare Richtlinie für die Vertragsparteien darstellt, zumal hier eine Hierarchie fehlt und auch Widersprüche zwischen den Anforderungen nicht ausgeschlossen sind.

Bemerkenswert ist schließlich, dass die Projektleitung nur in 4 Mustern ausdrücklich geregelt ist. Bei HE (§ 2), SN-S (Nr. 6.1 und 6.5) wird sie gemeinsam ausgeübt, wobei der Vorsitz bei HE vom Softwarebesteller eingenommen wird. Bei BA (§ 11 Abs.1) und SN-B (Nr. 1.2) wird sie dem Softwarehaus zugeordnet.

KO (§ 7 Abs.1), MA (§ 4), BA (§ 7 Abs.1 und 2⁵²²), SN-S (Nr. 7.1), SN-B (Nr. 8.1) und SÖ (§ 8 Abs.1⁵²³) enthalten eine Fixpreisvereinbarung. Die BVB (§ 7) lassen die Vergütungsfrage ebenso wie GL (Nr. 13), der allerdings verschiedene ausführliche Optionen für eine Festpreisvereinbarung oder eine aufwandsabhängige Vergütung anbietet, offen.

B. Beispiele aus der Praxis

Im Folgenden werden drei Verträge untersucht, die anlässlich tatsächlich durchgeführter Softwareentwicklungsprojekte geschlossen wurden⁵²⁴. An den ersten beiden waren jeweils ein Softwarehaus aus Mittelhessen und einmal ein Produktionsbetrieb in Österreich (nachfolgende MiO⁵²⁵) und zum anderen

⁵²² Hier findet sich eine ausgefeilte Regelung, die durch eine Bonus/Malus System (§ 19) ergänzt wird und zu einer Vergütungsspanne zwischen 88 und 144 % des ursprünglichen Preises führt.

⁵²³ Wobei, dem Prinzip des Lizenzvertrages folgend, je nach Lizenzmodell für die Vergütungsregelung mehrere Varianten mit teilweise fortlaufend zahlbarer Lizenzgebühr vorgeschlagen werden.

⁵²⁴ Dem Verfasser liegen alle Verträge, die ihm im Rahmen seiner Tätigkeit als Rechtsanwalt zugänglich wurden, im vollständigen Wortlaut vor. Aus Verschwiegenheitsgründen sind die Parteien und Produkte hier anonymisiert. Anders als in den USA ist es in Deutschland ungleich schwerer, empirisches Material zu erhalten. Die Verträge wurden - soweit erforderlich - der neuen Rechtschreibung angepasst.

⁵²⁵ Dieser Vertrag wurde 1995 abgeschlossen. Das Projekt wurde nach etwa 5 Jahren und nach Überwindung erheblicher Projektstörungen einschließlich des Einbeziehens sachverständiger und gerichtlicher Hilfe erfolgreich abgeschlossen. Das Softwarehaus wird als Auftragnehmer, der Softwarebesteller als Auftraggeber bezeichnet. Der Auftraggeber stammt aus der Baubranche und verfügt über mehrere Standorte. Er wollte seine vorhandene EDV auf ein neues System umstellen.

ein Produktionsbetrieb in Nordrhein-Westfalen (nachfolgend MiN⁵²⁶) beteiligt. Das dritte betrifft ein Softwarehaus in Südhessen und einen Softwarebesteller der öffentlichen Hand im Siegerland (nachfolgend SüS⁵²⁷).

I. Vertragsart

MiO gliedert sich in den hier untersuchten Hauptvertrag und sechs „*Beilagen*“, die z.B. Detailinformationen zu den Anforderungen des Auftraggebers (Beilage 1), technischen Voraussetzungen wie Endgeräte, Schnittstellen, Systemkomponenten (Beilage 2), Installation (Beilage 3), Programmierrichtlinien des Auftragnehmers (Beilage 5), Beispiele für Source-Code Kommentare (Beilage 6) enthalten. Der Vertrag ist bezeichnet als „*Werkvertrag über die Lieferung, Installation und Wartung einer Gesamtlösung für Vertrieb und Produktion von x*“. Die Grundregelung⁵²⁸ für den Vertragsgegenstand ist in „**3. Vertragsgegenstand**“ enthalten (Die Satznummerierung und die Nummerierung der Unterpunkte wurde zur besseren Zitierbarkeit ergänzt. Sie sind im Original nicht vorhanden).

„¹Gegenstand dieses Vertrages ist die Lieferung, Inbetriebnahme, Einführung und Wartung der Anwendungssoftware für den in der Präambel genannten Zweck, wobei einige Teilleistungen (...) nicht Vertragsgegenstand sind, wie sich aus den folgenden Detaildefinitionen ergibt.

²Insbesondere gehört zu den Pflichten des Auftragnehmers

- 1. die Definition einer lauffähigen Konfiguration aus Hardware, Software und Netzwerkverbindungen, die die beim Auftraggeber und seinen Händlern anfallenden Datenmengen und Anwendungen unter Verwendung der in Beilage 2 definierten Anzahl an Endgeräten und unter Betrieb der in Beilage 2 definierten Schnittstellen zu anderen Systemen bewältigen kann,*
- 2. die Definition der Schnittstellen zu anderen Systemen gemeinsam mit dem Auftraggeber,*

⁵²⁶ Der Vertrag wurde 1999 unterzeichnet. Der Softwarebesteller wurde durch ein Consultingunternehmen bei dem Vertragsentwurf und der Vertragsverhandlung unterstützt. Ob das Projekt erfolgreich durchgeführt wurde, ist nicht bekannt. Das Softwarehaus wird als Auftragnehmer, der Softwarebesteller als Auftraggeber bezeichnet. Die Softwarelösung sollte auf Basis der vom Auftragnehmer entwickelten Standardmodule an die individuellen Verhältnisse beim Auftraggeber angepasst werden.

⁵²⁷ Dieser Vertrag wurde im August 2001 unterzeichnet und das Projekt nach etwa zwei Jahren erfolgreich abgeschlossen. Das Softwarehaus wird als Auftragnehmer, der Softwarebesteller als Partner bezeichnet. Es handelte sich um die vollständig Neuentwicklung einer zum Vertrieb vorgesehenen Standardsoftware.

⁵²⁸ Unter 6. findet sich eine weitere Vereinbarung mit dem Titel „*Vertragsgegenstand Wartung*“, der diesbezügliche Details regelt.

3. die Prüfung der Vorgaben des Auftraggebers (Pflichtenheft) auf Sinnhaftigkeit und Machbarkeit,
4. die Anpassung bzw. Erweiterung der Standardsoftware an die Anforderungen des Auftraggebers wie in Beilage 1 spezifiziert und im Probetrieb der Vorversionen noch genauer zu definieren,
5. die Installation der Anwendungssoftware gemäß Beilage 3 inklusive der Anpassungen,
6. die Optimierung der zentralen Datenbanksysteme in Bezug auf Laufzeitverhalten und Inanspruchnahme von Ressourcen,
7. die Lieferung des Sourcecodes der Anwendungssoftware ohne die Sourcen für Zeichen- der CAD-Routinen,
8. die Unterstützung des Auftraggebers bei der Inbetriebnahme und Erfassung der Stammdaten,
9. die Mitwirkung an den Abnahmetests
10. die Lieferung der Dokumentation,
11. die Schulung des Personals des Auftraggebers,
12. die Wartung der Anwendungssoftware gemäß den Regelungen dieses Vertrages,
13. die Lieferung und Anpassung der Anwendungssoftware in der Produzentenversion und Händlerversion für die Firma X zu diesem Vertrag vergleichbaren Bedingungen und auf besonderen Auftrag. Die Lieferung des Sourcecodes ist in diesem Fall gesondert zu diskutieren.

³Die genannten Pflichten sind Hauptleistungspflichten und bilden mit Ausnahme der Wartung eine unteilbare Leistung.

⁴Im Fall der Zerstörung der Software wird der Auftragnehmer - bei aufrechtem Wartungsvertrag - innerhalb von einem Werktag eine Kopie der jeweiligen Software kostenlos nachliefern (DFÜ bzw. Versand/ Kurier) oder innerhalb von zwei Werktagen die notwendigen Arbeiten zur Installation der Software beginnen.“

Aus S. 2 Nr. 3 ergibt sich, dass das Pflichtenheft vom Softwarebesteller vorgelegt wurde. Die hier festgeschriebene Prüfpflicht auf „Sinnhaftigkeit“ macht dabei wenig Sinn, solange es an einem Bezugsrahmen fehlt. Selbst soweit hier sehr allgemein von den Interessen des Softwarebestellers ausgegangen wird, so ergibt sich daraus infolge der Subjektivität kein geeigneter Kontrollmaßstab. Dabei stellt das geprüfte Pflichtenheft nach der unmittelbar folgenden Nr. 4 keine vollständige Leistungsbeschreibung dar. Es bedarf vielmehr noch einer genauen Definition durch einen Probetrieb der Vorversionen.

Min besteht aus einem als „Werkvertrag“ bezeichneten Hauptvertrag und vier Anlagen, nämlich einem „Source-Code Nutzungsvertrag“, einem „Pflichtenheft“, einem „Abweichungsbericht mit Priorisierung (Stand X)“ und den „Bedingungen des Angebotes vom X“.

Der Vertragsgegenstand ist unter „**1. Vertragsgegenstand**“ geregelt:

„1.1 Gegenstand dieses Vertrages ist die vollständige und funktionsfähige Erstellung eines auf die konkreten Bedürfnisse des AG zugeschnittenen EDV-Systems als Produktions- und Händler-Lösung. Der Auftrag umfasst Software, Installation, Einweisung, Schulung und Wartung. Geschuldet wird ein in sich geschlossenes System, bei dem Software, Anpassungsprogrammierung, Einrichtungsarbeiten sowie Dienstleistungskomponenten aufeinander abgestimmt sind und den vertraglichen Anforderungen des AG, die dem AN im Pflichtenheft dargestellt sind, Rechnung tragen.

Das System wird gemäß beigefügtem Angebot realisiert. Maßgeblich für den Vertragsinhalt sind in jedem Fall die Anforderungen gemäß Pflichtenheft. Das Pflichtenheft formuliert die Geschäftsprozesse und funktionalen Anforderungen des AG, die es mit der Software abzubilden gilt. Der Vertrag wird in mehreren Phasen nach den im Terminplan genannten Meilensteinen realisiert. Zahlungen für termingerecht erbrachte und abgenommene Leistungen können auch bei vorzeitigem Vertragsende nicht zurückverlangt werden.

1.2 Die Anforderungen des AG wurden im Vorfeld durch ein Pflichtenheft formuliert. Zu diesem Pflichtenheft wurde seitens des AN ein Abweichungsbericht erstellt. Im Abweichungsbericht nicht erwähnte Punkte des Pflichtenheftes werden als erfüllbar betrachtet. Aufbauend auf diesem Abweichungsbericht ist vom AG zu den einzelnen Punkten die terminliche Priorisierung eingetragen. Anforderungen, die nicht Bestandteil dieses Vertrages sind, sind dort explizit durch den AG als solche gekennzeichnet. Die Erfüllung der in dem Abweichungsbericht zugesagten Anforderungen beziehen sich auf die Formulierungen des Pflichtenheftes. Der AN sichert dem AG zu, dass die im Pflichtenheft / Abweichungsbericht dargestellten Leistungen durch die Standardsoftware und Anpassungen zu den terminlich vereinbarten Meilensteinen erbracht werden.“

So eindeutig, wie es die Vertragsbezeichnung bei MiO und MiN aussagt, fällt die Vertragseinordnung nicht aus, da sowohl MiO (S. 2 Nr.11) als auch MiN (Nr. 1.1) Schulungsleistungen erfassen, die allgemein als Dienstleistungen betrachtet werden. Beide Verträge machen deutlich, dass es dem Softwarebesteller auf eine Betreuung über den gesamten Lebenszyklus der Software und nicht nur auf die Erstellung eines Programms ankommt. Betont wird dies bei MiO durch S. 3, wonach alle Pflichten bis auf die Wartung eine unteilbare Leistung darstellen sollen. Ein weiterer Hinweis darauf ist auch der Präambel bei MiO zu entnehmen:

„Der Auftraggeber beabsichtigt, die EDV-Unterstützung der Vertriebs- und Produktionstätigkeiten seines und der verbundenen Unternehmen (z.B. X) sowie die Vertriebsaktivitäten seiner Händler bis spätestens März 1996 auf eine neues System umzustellen, und wurde diesbezüglich vom Auftragnehmer beraten.“

Auf diese Präambel wird in Nr. 3 S.1 als „Zweck“ der Leistungen des Softwarehauses Bezug genommen.

Der Wortlaut bei MiN lässt erkennen, was im Vorfeld der Vertragsunterzeichnung geschehen ist. Nachdem das Softwarehaus ein Angebot erstellt hatte, wurde auf dessen Basis (durch den Softwarebesteller) ein „Pflichtenheft“ erarbeitet.

Zu diesem hat das Softwarehaus einen „Abweichungsbericht“ vorgelegt, der - unter Einschränkung der Anforderungen des Pflichtenhefts - die zugesagten Leistungsmerkmale festlegt. Das insofern irreführend als „Pflichtenheft“ bezeichnete Dokument stellt eher eine „Wunschliste“ des Softwarebestellers dar, die das Softwarehaus auf Realisierbarkeit prüfte (*„Im Abweichungs-bericht nicht erwähnte Punkte des Pflichtenhefts werden als erfüllbar betrachtet“*).

Angebot, „Pflichtenheft“ und Abweichungsbericht bilden hier zusammen die Grundlage für die Beschreibung der Leistungspflicht, wobei der Geltungsvorrang, zumindest dem zeitlichen Zusammenhang nach, beim Abweichungsbericht liegt. Auch der zusätzliche Hinweis unter „**10.**

Schutzpflichten“

„10.3 Der AN hat das vom AG erstellte Pflichtenheft überprüft und keine Bedenken im Hinblick auf Funktions- und Einsatzfähigkeit gem. Nr. 1 des Vertrages festgestellt.“

ändert an der insoweit fehlenden endgültigen Klarheit nichts, denn als Ergebnis der Prüfung wurde ein Abweichungsbericht erstellt, der eine Zusammenstellung von gerade nicht erfüllbaren Anforderungen darstellt.

Die hier fehlende klare Priorisierung findet sich bei dem als „Softwareentwicklungsvertrag“ bezeichneten Dokument SüS unter „**1.**

Gegenstand des Vertrages“ (Hervorhebungen im Original):

„Gegenstand des Vertrages ist die Durchführung des folgenden Entwicklungsvorhabens durch den Auftragnehmer für den Partner: „Planung, Evaluierung und Realisierung der Software X“, (im Folgenden „Softwareerstellungvertrag“).

Der Gegenstand des Softwareerstellungsvertrages und der genaue Umfang der vom Auftragnehmer durchzuführenden Arbeiten sowie gegebenenfalls die Mitwirkungspflicht des Partners ergibt sich aus

- a) **dem Pflichtenheft** „X“ mit dem Stand vom 23.03.2001
 - b) **dem Angebot** des Auftragnehmers vom 28.05.2001
 - c) der Auftragsbestätigung des Partners vom 25.07.2001
 - d) der „**Ergänzungen zum Pflichtenheft**“ mit Stand vom 05.07.2001.
- Maßgeblich ist das Pflichtenheft.“*

Auch hier wurden vor Vertragsschluss Schriftstücke ausgetauscht, wobei sich aus den Datumsangaben interessante Schlussfolgerungen ergeben. Danach hat zunächst das „*Pflichtenheft*“ (als Grundlage für das Einholen von Angeboten und damit durch den Softwarebesteller erarbeitet) vorgelegen. Auf dessen Basis hat das Softwarehaus ein Angebot abgegeben, woraufhin Änderungen am „*Pflichtenheft*“ vorgenommen und durch den Softwarebesteller eine Auftragsbestätigung übersandt wurde.

SüS sieht eine ausschließliche (7.1), MiO (Nr.22) und MiN (Nr. 1 des Source-Code Nutzungsvertrages) sehen eine einfache, nicht übertragbare Lizenz vor, wobei MiN für kundenspezifische Auftragsprogrammierung ein ausschließliches Verwertungsrecht beim Softwarebesteller begründet (10.1).

II. Bestandteile der Lieferung

MiO verlangt unter Nr. 3 S.1 (s.o.) die Lieferung der Anwendungssoftware und unter S. 2 Nr. 10 die Lieferung der Dokumentation sowie des Sourcecodes der Anwendungssoftware (S. 2 Nr. 7 und Nr. 9).

„Teil des Vertragsgegenstandes ist die Lieferung des Sourcecodes aller Programme mit Ausnahme der Zeichenroutinen und CAD-Module, sowie dessen Updates. Der Auftraggeber hat das Recht, den Sourcecode ausschließlich im Rahmen des im Punkt Immaterialgüterrechte beschriebenen Nutzungsbereiches zu nutzen.“

Der Lieferung des Sourcecodes, als Übertragung des eigentlichen Know-How, wird besondere Bedeutung beigemessen, wie sich aus „**Nr. 10 Terminplan**“ ergibt. Dort heißt es in Absatz 3:

„Die Lieferung des Sourcecodes erfolgt jeweils Zug um Zug gegen Zahlung des bei Abnahme der Softwarelieferung fälligen Betrages, im

Falle der Abnahme bei X auch gegen Rückstellung der Bankgarantie für die Vorauszahlung.“

Diese Lieferpflicht des Sourcecode darf hier insofern als ungewöhnlich angesehen werden, als das Softwarehaus auch für die Wartung verantwortlich sein soll und der Softwarebesteller kein ausschließliches Nutzungsrecht erhält. Sie wird verständlich, wenn der Softwarebesteller über eine eigene Entwicklungsabteilung verfügt, die sich an Wartung und Pflege der Software beteiligt, ohne alleine dafür verantwortlich sein zu können oder zu wollen.

Durch die ausführliche Pflichtdefinition zur Dokumentationslieferung wird dieser Gedanke bestätigt. „**23. Dokumentation**“ enthält dazu folgende Regelung:

„Teil des Vertragsgegenstandes bei Lieferung und Wartung ist die Lieferung und laufende Aktualisierung Benutzer- und technischer Dokumentation für alle gelieferten EDV-Komponenten sowie die Lieferung eines Testdatenbestandes.

Die Benutzerdokumentation besteht mindestens aus einem Benutzerhandbuch und der kontextintensiven Online-Hilfe.

Benutzerhandbuch und Handbuch für Installation und Administration haben alle für die laufende Arbeit notwendigen Abläufe so zu beschreiben, dass sie für eine eingeschulte Person verständlich sind. Daneben hat die Dokumentation alle typischen und vorhersehbaren Fehler- und Mängelsituationen darzustellen und deren Behebung zu beschreiben.

Das Handbuch für Systembediener muss neben den wichtigsten Teilen des Systemhandbuchs insbesondere einen Arbeitsplan gegliedert nach täglichen, wöchentlichen, monatlichen und jährlichen Vorgängen enthalten sowie alle Aktionen im Fehlerfall (Sicherung, Restore von Dateien und Datenbanken, ...) beschreiben.

*Die technische Dokumentation besteht aus dem Data Dictionary (Beschreibung der Datensätze und Datenfelder mit Format und Kommentar), Kommentaren im Sourcecode, wie sie in den beiliegenden Beispielen (Beilage 6) exemplarisch dargestellt sind, und aus der Schnittstellenbeschreibung zu den nicht im Sourcecode gelieferten Modulen.
(...)“*

Erkennbarer Schwerpunkt der Dokumentationslieferung ist es, die Verwertbarkeit der Unterlagen in der Praxis sicherzustellen. Für die

technische Dokumentation akzentuiert der letzte Absatz dies, indem er die konkrete Gestaltung der Kommentare nicht dem Softwarehaus überlässt, sondern eine eigene Vorgabe in Form von Beispielen liefert. Der Entwickler soll auf diese Weise daran gehindert werden, die Erläuterung seiner Arbeit in für Dritte unverständlicher Form vorzunehmen⁵²⁹.

Bei MiN ist die Lieferpflicht der Software nicht ausdrücklich benannt. Unter der etwas irreführenden Überschrift „**12. Dienstleistungen**“ heißt es:

„Der AN stellt dem AG nach Vertragsschluss ein lauffähiges Standardsystem mit Datenbank und 10 PC-Arbeitsplätzen zum Probetrieb zur Verfügung, damit das System kennen gelernt, voreingestellt und mit Stammdaten bestückt werden kann.“

Eine Lieferpflicht ergibt sich dann aber aus „**13. Fehlerbeseitigung**“ und hier unter „**13. 4 Updates und Upgrades**“:

„Gegenstand des Wartungsvertrages ist ferner die kostenlose Lieferung neuer, verbesserter und erweiterter Programmversionen (Updates und Upgrades) der Software. (...)“

Die Lieferung der Dokumentation ist gemäß „**11. Dokumentation**“ vorgesehen:

„11.1 Die Dokumentation ist einfach zu erstellen oder per EDV aufzurufen - sie darf zum internen Gebrauch beliebig vervielfältigt werden - und umfasst:

11.1.1 Die Softwaredokumentation (Programmbeschreibungen, Handbücher), einschließlich der angepassten und erweiterten Programme - die Dokumentation ist so detailliert zu erarbeiten, dass anhand der Handbücher eine Programmbedienung durch den geschulten Mitarbeiter möglich ist -,

11.1.2 Die detaillierte Dokumentation der Gesamtkonfiguration. Zur Gesamtkonfiguration gehört das gesamte Ablageschema der Daten inkl. aller Codierungen und Bedeutungen. Soweit die Dokumentation nicht durch den entsprechenden Ausdruck der Konfigurationsdateien sichergestellt werden kann, übergibt der AN dem AG das gesamte Ablageschema zusätzlich in Papierform. Über Änderungen des Ablageschemas der Daten informiert der AN den AG unverzüglich.

11.2 Die Dokumentation ist spätestens 3 Monate nach der Endabnahme vorzulegen und wesentlicher Bestandteil der vertragliche Hauptleistung.“

⁵²⁹ Diese Gefahr ist u.a. deswegen gegeben, weil die Dokumentation der Codierung sich in Entwicklerkreisen keiner großen Beliebtheit erfreut.

Hier wird erneut deutlich, dass der Softwarebesteller vor allem eine *Programmbedienung* und nicht eine eigene Programmpflege im Sinn hat. Der nicht klar ersichtliche Inhalt der Dokumentation zur „*Gesamtkonfiguration*“ dürfte auch vor allem dazu dienen, ein Verständnis des Softwarebestellers dafür zu ermöglichen, wie der Umgang mit den Daten vom Programm gehandhabt wird. Dies kann dazu beitragen, die Abhängigkeit vom Wissensmonopol des Softwarehauses zu verringern.

Diese Absicht wird im Source-Code Nutzungsvertrag unterstrichen. Das Softwarehaus wird in der Bestimmung „**4. Source Code**“ zur Einräumung eines nicht ausschließlichen, nicht übertragbaren Nutzungsrechts verpflichtet

„4.3 Der AG erhält das zeitlich und räumlich unbeschränkte Recht, den Quellcode auf sämtliche Arten zu nutzen, ihn insbesondere zu übersetzen, zu bearbeiten, das Arrangement und andere Umarbeitungen des Quellcodes zu verwerten sowie die erzielten Ergebnisse zu vervielfältigen. Zu diesem Zweck übergibt der AN neben dem gesamten eigenen Source-Code (inkl. Projektdateien, Einstellungen) dem AG alle Angaben zu notwendigen Hilfsmitteln (Libraries, Compiler, usw.) zur Erzeugung der Binaries. (Anm.: Source-Code und Quellcode sind Synonyme. Er repräsentiert das Programm in einer höheren Programmiersprache, das vom Computer jedoch erst verstanden wird, wenn es kompiliert, d.h. in für Computer lesbare Binärprogramme – Binaries - übersetzt wurde). Dies gilt ebenfalls für alle Folgeupdates und Änderungsdokumentationen. Zur Bewertung werden einige typische Source-Code Dateien vorab übergeben.“

Bei SüS ist die Lieferung der Software unter „**9. Abnahme**“ vorgesehen.

„9.1. Der Auftragnehmer liefert die Rohversion, die vorläufige Endversion und die Endversion der Software auf einem festen Datenträger (CD-ROM, Disketten oder vergleichbares mit vorhandenen Mitteln des Partners lesbares Speichermedium). Zwischenversionen können per DFÜ (Anm.: = Datenfernübertragung) übertragen werden.“

Dokumentation und Quellcode sind in „**12. Dokumentation und Quellcode**“ geregelt.

*„12.1. Der Auftragnehmer übergibt mit jeder Version (insbesondere Rohversion, vorläufige Endversion und Endversion) den Quellcode des Programmes an den Partner.
12.2. Der Auftragnehmer liefert neben dem ablauffähigen Programm und dem Quellcode eine Programmdokumentation. Diese erläutert den Programmcode nach Maßgabe des allgemeinen technischen Standards. Sie ist in ihrem Umfang so zu bemessen, dass ein der Programmiersprache*

kundiger Mitarbeiter prinzipiell den Aufbau und die Arbeitsweise der Software nachvollziehen und bearbeiten kann. Die Programmdokumentation kann auch zum Bestandteil des Quellcodes gemacht werden.“

Hier zeigt sich die Ausrichtung des Vertrages, wie bereits durch die Einräumung des ausschließlichen Nutzungsrechtes dokumentiert, deutlich. Der Softwarebesteller möchte in die Lage versetzt werden, das eigens für ihn erstellte Programm ohne fremde Hilfe, v.a. der des Softwarehauses, verändern und weiterentwickeln zu können. Eine Betreuung über den gesamten Lebenszyklus, insbesondere also die Wartung nach der Erstellung ist nicht vorgesehen.

III. Spezifikationen und Leistungsmaßstab

Die Spezifikation der Leistung ergibt sich bei MiO aus dem Pflichtenheft und dem - noch durchzuführenden - Probetrieb der Vorversionen (siehe oben I.). MiN bezieht sich dafür auf das Pflichtenheft, den Abweichungsbericht und das Angebot des Softwarehauses (s.o. I.). Bei SüS erstreckt sich die Bezugnahme auf Pflichtenheft, Angebot, Auftragsbestätigung und sog. Ergänzungen (s.o. I.). In den letzten beiden Fällen wird dem Pflichtenheft die vorzugsweise Geltung eingeräumt. Dadurch wird sehr anschaulich, wie das Bemühen um eine Fixierung der näheren Leistungsbestimmung durch den dynamischen Prozess, in dem sie zustande kommt, erschwert wird.

Ein Versuch, dieser Unsicherheit zu begegnen, findet sich bei MiO im Wege eines extensiv formulierten Leistungsmaßstabes (Absatz- und Unternummerierungen wurden ergänzt):

„15. Qualitätsanforderungen

- (1) *Auftraggeber und Auftragnehmer stimmen überein, dass an die Entwicklung von Software hohe Qualitätsanforderungen gestellt werden, es aber dennoch nicht möglich ist, Software zu entwickeln, die mit Sicherheit frei von Fehlern ist.*
- (2) *Der Auftraggeber erfüllt die Qualitätsanforderungen,*
 1. *dass bei EDV-gestützten Vorgängen an allen in Beilage 2 definierten Anlagen bei typischer Lastsituation zur Tageszeit der größten Last in 95% aller Fälle Antwortzeiten unter drei Sekunden auftreten. Es ist also durchaus möglich, dass manche Benutzeraktionen Antwortzeiten von über drei Sekunden zur Folge haben,*
 2. *dass das System so gestaltet ist, dass es leicht erlernt und bedient werden kann,*

3. dass auch die vorhandenen HP-UX-Systeme des Auftraggebers verwendet werden können,
- (3) Der Auftraggeber wird Software liefern, die maßgeblich die folgenden Anforderungen erfüllt (in 95 % aller Fälle):
1. die nicht nur auf Funktionalität sondern auch auf Verhalten in Grenzfällen (Fehleingaben, Anzahl gleicher Transaktionen, Datenmengen) getestet wurde und
 2. für jedes Formular eine kontextintensive Hilfe anbietet,
 3. für jede Eingabe von Werten aus einem vordefinierten Wertebereich eine Liste von zulässigen Werten anzeigt,
 4. gegen Fehlbedienung durch Verwendung von Wertebereichsprüfungen und Integritätsregeln abgesichert ist bzw. die Benutzer bei kritischen Eingaben auf die möglichen Folgen hinweist,
 5. deren Sourcecode ausschließlich den dokumentierten Standardsprachumfang der Programmiersprache von Paradox Object PAL bzw. Visual C++ bzw. definierte zusätzliche Werkzeuge verwendet,
 6. zu deren Installation keine Änderungen am Maschinencode des Betriebssystems notwendig sind,
 7. in der Regel Zahlenwerte wie Prozentwerte, Wertgrenzen u.ä. aus einer leicht wartbaren Tabelle außerhalb des Programmcodes gelesen werden, so dass bei vielen zu erwartenden nicht die Programmlogik betreffenden Änderungen in den Anforderungen an die Anwendung nur Tabellen, aber keine Zahlenwerte in Programmen geändert werden müssen.
 8. die gemäß den Programmierrichtlinien des Auftragnehmers (Beilage 5) erstellt und gewartet wird,
 9. die bei der Produktionssoftware vom Konzept einer relationalen Datenbank ausgeht, auf die mit der Sprache SQL zugegriffen wird,
 10. die nicht nur vom Programmierer getestet wird,
 11. bei der Änderungen und anwendungsspezifische Anpassungen so durchgeführt werden, dass solche Änderungen klar erkennbar sind und die Versionsfähigkeit nicht verloren geht,
 12. die Transaktionen auf zentralen Datenbanken (ORACLE) mit Beginn- und Endbefehl für das Datenbanksystem transparent macht,
 13. die die logische Zusammengehörigkeit von Tabelleneinträgen durch explizite Aufnahme der foreign keys transparent macht,
 14. die in der Produktionssoftware keine Mechanismen enthält, die die Benutzung durch den Anwender zeitlich begrenzen oder den Wechsel zwischen verschiedenen Arbeitsplätzen erschweren (Kopierschutz),
 15. die die Übernahme von Daten in die vorhandenen PC-Systeme des Auftraggebers zwecks Auswertung mittels Tabellenkalkulation ermöglicht.“

Interessant ist hier zunächst die Aussage zur unvermeidbaren Fehlerhaftigkeit von Software in Abs. 1. Die unter Abs. 2 Nr. 1 und 2 aufgeführten Punkte können als Generalklauseln zur Sicherung einer effektiven Nutzbarkeit verstanden werden. Die im Abs. 3 genannten Punkte stellen im Anschluss daran Detailkriterien für eine dementsprechende Umsetzung auf. Wie sich aus der Regeln des Abs. 3 Nr. 7 ergibt, werden „viele“ Änderungen erwartet, weshalb ein einfacher Zugriff auf zu ändernde Softwareteile gewährleistet werden soll. MiO orientiert sich an dem

Grundgedanken, die Schwierigkeit bei der Bestimmbarkeit des „Was“ durch ein Regeln des „Wie“ zu kompensieren.

MiN regelt den Leistungsmaßstab in „**Nr. 6 Zusicherungen**“.

„6.1 Der AN erklärt sich bereit, im Falle von auftretenden Problemstellungen, die das Gesamtsystem (Hard- und Software) betreffen, auf Anfrage durch den AG über das normale Maß hinaus im Rahmen der vorhandenen Kapazitäten gegen Vergütung bei der Lösung der Problemstellung zu helfen und zu unterstützen.

6.2 Der AN sichert zu, dass die gesamte Software 2000 und EURO-fähig ist.

6.3 Der AN sichert zu, dass die vom System erzeugten Daten korrekt sind, soweit der AG die Stammdaten ordnungsgemäß angelegt hat und keine Eingriffe in die vom AN gelieferten Systemkomponenten vorgenommen wurden. Der Nachweis für fehlerhafte Berechnungen muss vom AG geführt werden. Für vom AG vorgenommene Eingriffe übernimmt der AN keine Haftung.

(...)

6.6 Die Softwarepflege erfolgt in der Form, dass Updates auf dem zentralen Applikationsserver installiert werden und die jeweiligen Arbeitsplätze selbständig die neuen Versionen laden.

6.7 Die Programm- und Datenstrukturen sind so ausgelegt, dass der AG Erweiterung an den Datentabellen vornehmen kann (z.B. Statistikfelder), ohne dass die Programmfunktionalität darunter leidet und die Update-Fähigkeit eingeschränkt ist.“

Stammen Software und Hardware - wie im Regelfall - nicht aus einer Hand, so ist dem Softwarebesteller daran gelegen, gleichwohl eine Betreuung des Gesamtsystems sicherzustellen. Dies spiegelt sich in 6.1 wieder, wobei die Pflichten des Softwarehauses hier durch die Begrenzung auf vorhandene Kapazitäten und gegen Vergütung nicht unerheblich relativiert werden. Weiter soll durch konkrete Benennung bestimmten Risiken für den Softwarebesteller vorgebeugt werden (Nr. 6.2). Erneut findet sich in Nr. 6.7 eine im Hinblick auf Änderungen von Teilen der Software konzipierte Klausel.

IV. Besondere Pflichten des Softwarehauses

SüS legt besonderen Wert darauf, dem Softwarebesteller eine Kontrolle der Tätigkeit des Softwarehauses zu ermöglichen:

„2.3 Der Auftragnehmer wird dem Partner auf Wunsch jederzeit Einblick in die jeweils vorliegenden Arbeitsergebnisse geben, alle sonstigen gewünschten Auskünfte erteilen sowie Beauftragten des Partners während

der bei dem Auftragnehmer üblichen Arbeitszeit Zutritt zu den Räumen gewähren, in denen die Softwareentwicklung durchgeführt wird.“

Eine weitergehende Variante findet sich auch bei MiN - etwas „versteckt“ - im Source-Code Nutzungsvertrag unter „**6. Kontrollrechte**“

„Die Parteien haben das Recht, jederzeit ohne Vorankündigung bei dem jeweiligen Vertragspartner die korrekte Anwendung des Vertrages zu überprüfen.“

Zur Sicherung der termingerechter Fertigstellung verankert MiN v.a. Informationspflichten unter „**4. Lieferung**“:

„4.2 Der AN ist verpflichtet, eine voraussichtlich wesentliche (länger als fünf Werktagen anhaltende) Verzögerung dem AG unverzüglich schriftlich anzuzeigen, nachdem für den AN die Verzögerung erkennbar wird. Sofern der AG seine Zustimmung nicht innerhalb von sieben Werktagen verweigert, verlängern sich die Folgetermine um die Dauer der angezeigten Verzögerung. Verweigert der AG seine Zustimmung, so verständigen sich die Vertragspartner über die weitere Vorgehensweise. Kann keine Einigung erzielt werden, so ist nach Maßgabe der Nr. 22 (Anm.: Nr. 22 enthält eine Schiedsklausel) zu verfahren.

4.3 Entsteht eine wesentliche Verzögerung durch höhere Gewalt oder ist sie durch den AG zu vertreten, so zeigt der AN dies ebenfalls unverzüglich unter Hinweis auf die voraussichtliche Dauer der Verzögerung an. In diesem Fall verlängern sich die Termine jeweils um die Dauer der angezeigten Verzögerung.

Der höheren Gewalt stehen Streik, Aussperrung oder ähnliche Umstände, von denen der AN mittelbar oder unmittelbar betroffen ist, gleich.

(...)

4.6 Kommt der AN mit dem Testbetriebsende (Abnahmereife) schuldhaft nach Maßgabe der Bestimmungen dieses Vertrages in Verzug, so verpflichtet er sich, für jede Woche der Terminüberschreitung eine Vertragsstrafe von 2 % des Pauschal festpreises, insgesamt aber höchstens 10 % des Pauschal festpreises an den AG zu zahlen. Weitergehende Schadensersatzansprüche des AG sind ausgeschlossen, es sei denn, die Haftung ergibt sich aus Nr. 9 (Anm.: Nr. 9 enthält eine allgemeine Haftungsregel) dieses Vertrages.“

Nr. 4.2 betrifft, wie sich im Gegenschluss aus 4.3 ergibt, vom Softwarehaus zu vertretende voraussichtliche Projektverzögerungen. Es wird der etwas hilflos wirkende Versuch unternommen, den Terminplan durch ein der Verzögerungsmittelteilungs anschließendes Verfahren zu retten. Eine Verweigerung der Zustimmung ändert an den Umständen (d.h. den Gründen und Auswirkungen der Verzögerung) nichts. Möglicherweise ist die Vertragsstrafe bei Verzögerung in Nr. 4.6 wirkungsvoller.

Bei MiO finden sich unter **28. Informationspflichten** folgende Regelungen:

„(...) Der Auftragnehmer wird dem Auftraggeber spätestens am 10. jeden Monats Bericht über den Projektfortschritt im vergangenen Monat geben. Der Auftragnehmer wird sofort nach Vertragsschluss dem Auftraggeber eine Liste seiner bei der Vertragsdurchführung beteiligten Mitarbeiter zusenden und dem Auftraggeber jede personelle Veränderung bei den beteiligten Mitarbeitern mitteilen. (...)“

Die übliche Berichterstattung wird hier um eine gesonderte Informationspflicht bezüglich der Mitarbeiter beim Softwarehaus erweitert. Ein Einwirkungsrecht auf Personalentscheidungen ist damit nicht verbunden. Der Softwarebesteller hat durch die Übersicht über den Personalbestand beim Vertragspartner aber ein „Frühwarnsystem“ an der Hand, das ihm helfen kann, mögliche Projektstörungen vorherzusehen.

V. Besondere Pflichten des Softwarebestellers

Besondere Pflichten des Softwarebestellers beschränken sich in SüS auf eine kurze Regelung unter **2. Durchführung der Arbeiten**:

„2.1 Der Partner wird auf Anfrage alle für die Durchführung der X-Software erforderlichen Auskünfte rechtzeitig erteilen. (...)“

Dies wird vor allem daran liegen, dass im Ergebnis die Entwicklung einer Standardsoftware Vertragsgegenstand und eine Integration in eine vorhandene Anwendungsumgebung nicht erforderlich ist. Damit entfällt auch die Notwendigkeit, Ressourcen in anderer Form als der Lieferung von Informationen zur Verfügung zu stellen.

Bei MiO und MiN verhält sich dies anders, weshalb hier ausführlichere Regelungen anzutreffen sind.

Bei MiO heißt es unter **„8. Pflichten des Auftraggebers“** (die Absatznummerierung wurde wieder ergänzt):

„(1) Der Auftraggeber wird dem Auftragnehmer alle notwendigen Informationen zur Verfügung stellen, soweit der Auftragnehmer diese

fristgerecht anfordert und es der laufende Betrieb erlaubt. Eine Verzögerung der Bereitstellung von Informationen verhindert die kurzfristige Fehlerbeseitigung bzw. Lokalisierung.

(2) Der Auftraggeber wird dem Auftragnehmer zur Anpassung des Vertragsgegenstandes an das Datenbanksystem ORACLE eine Lizenz dieses Datenbanksystems leihweise auf Dauer dieses Vertrages zur Verfügung stellen, soweit ORACLE diesem Vorgang zustimmt.

(3) Sollte ORACLE berechtigterweise nicht zustimmen, erwirbt der Auftraggeber für den Einsatz beim Auftragnehmer die kleinstmögliche ORACLE-Lizenz (z.B. für Novell), die dem Auftragnehmer eine Programmentwicklung gestattet, und stellt sie gemäß obigem Punkt zur Verfügung. Der Auftragnehmer wird anstreben, ORACLE-Business-Partner zu werden.

(4) Die Erstellung der Druckroutinen für Auftragspapiere wird nach den Anweisungen und Vorlagen des Auftragnehmers durch den Auftraggeber erfolgen.

(5) Der Auftraggeber wird für alle vertragsgegenständlichen Installationen der Software eine zentrale Verwaltungsstelle einrichten, die bei Problemen als First Level Support fungieren wird und neue Versionen (Upgrades und fehlerbereinigte Versionen) der Software an alle Systeme verteilen wird. Anfragen an den Auftragnehmer werden nur von diesem First-Level-Support oder von X getätigt.

(6) Die Schulung des Anwenderpersonals des Auftraggebers geschieht durch das Personal des Auftraggebers.

(7) Die Verteilung und Pflege der vom Auftragnehmer gelieferten Versionen der Händlersoftware geschieht durch den Auftraggeber.

(8) ...“

Der Softwarebesteller hat hier offenbar eigene Kompetenz oder soll sie jedenfalls erwerben und dann in seinem Betrieb für Aufgabenbereiche wie Schulung und Verteilung der Software zuständig sein. Eine Verquickung der Arbeitsbereiche ergibt sich aus Abs. 4, wonach das Softwarehaus „nach den Anweisungen und Vorlagen“ des Softwarebestellers tätig werden soll.

Die bereits bei SüS angetroffene Pflicht zur Informationslieferung wird auch hier als Holschuld begründet und darüber hinaus davon abhängig gemacht, dass es „der laufende Betrieb erlaubt“. Der anschließende Satz ist als Feststellung formuliert und allenfalls als Auslegungshilfe für den Fall hilfreich, in dem der Softwarebesteller die Lieferung der Information unterlässt. Konsequenz wäre es insoweit, an das Versäumnis einer Mitteilungspflicht konkrete Folgen zu knüpfen, wie beispielsweise eine Beweislastzuweisung zum Softwarebesteller für die Frage des Verschuldens von Leistungsverzögerungen.

Bemerkenswert ist hier auch der in Abs. 5 anzutreffende Hinweis, wonach die Vertragspartner davon ausgehen, es würden - gleichsam im Alltagsgeschäft - fehlerbereinigte Versionen vom Softwarehaus geliefert.

Bei MiN werden Pflichten des Softwarebestellers in „7.

Mitwirkungspflichten des AG“ formuliert:

„7.1 Der AG verpflichtet sich, die Tätigkeiten des AN zu unterstützen. Insbesondere schafft der AG unentgeltlich alle Voraussetzungen im Bereich seiner Betriebssphäre, die zur ordnungsgemäßen Durchführung des Auftrages erforderlich sind. Zu diesen Voraussetzungen zählen u.a., dass der AG

- Arbeitsräume für die Mitarbeiter des AN einschließlich aller erforderlichen Arbeitsmittel nach Bedarf ausreichend zur Verfügung stellt,*
- eine Kontaktperson benennt, die den Mitarbeitern des AN während der vereinbarten Arbeitszeit zur Verfügung steht; die Kontaktperson ist ermächtigt, Erklärungen abzugeben, die im Rahmen der Fortführung des Auftrages als Zwischenentscheidung notwendig sind. Es handelt sich bei dieser Person um X, der im übrigen auch neben der Geschäftsführung die Projektleitung wahrnimmt,*
- den Mitarbeitern des AN jederzeit Zugang zu den für ihre Tätigkeit notwendigen Informationen verschafft und sie rechtzeitig mit allen erforderlichen Unterlagen versorgt,*
- im Falle von Programmierarbeiten Rechnerzeiten (inkl. Operating), Testdaten und Datenerfassungskapazitäten rechtzeitig ohne gesonderte Kostenberechnung und in ausreichendem Umfang zur Verfügung stellt.*
- Der AG steht dafür ein, dass die im Rahmen des Auftrages vom AN gefertigten Berichte, Organisationspläne, Entwürfe, Zeichnungen, Aufstellungen und Berechnungen nur für seine Zwecke verwendet werden. Soweit an den Arbeitsergebnissen des AN Urheberrechte entstanden sind, verbleiben dieselben bei dem AN.*

7.2 Der AG stellt sicher, dass die Anwendungsumgebung den bei Vertragsabschluss vereinbarten und im Pflichtenheft oder Abweichungsbericht definierten Vorgaben entspricht und gemäß den übereinstimmend fortgeschriebenen Anforderungen ausgelegt ist.

(...)“

(7.3 siehe sogleich unten)

Die Verpflichtung zur Mitwirkung ist hier erheblich deutlicher akzentuiert als bei MiO. Der Generalklausel des Nr. 7.1 S.1 folgen, nicht abschließend, eine Mehrzahl von einzelnen Pflichten. Hier wird das Bereitstellen von Ressourcen (7.1, erster und dritter Spiegelstrich und 7.2) sowie die rechtzeitige und uneingeschränkte Informationslieferung (7.1, dritter Spiegelstrich) verlangt. Interessant ist die Zuweisung der Verantwortlichkeit für die Koordination und Überwachung des Change-Managements zu dem Projektleiter des Softwarebestellers (dazu sogleich).

VI. Veränderungen (Change Management)

Die aus einem Teil der Muster ersichtliche Regelungsdichte bezüglich des Change-Managements findet sich in den Praxisbeispielen (nur) ansatzweise wieder.

Die Formulierungen bei MiN und SüS sind sich sehr ähnlich. Bei SüS heißt es unter 7.3:

„10. Zusatzleistungen und Modifikationen

Verlangt der Partner eine nicht im Pflichtenheft enthaltene Leistung, so wird der Auftragnehmer die Anfrage kurzfristig beantworten. Macht ein Änderungsverlangen eine umfangreichere Prüfung erforderlich, wird der Auftragnehmer dies und die mit der Prüfung ggf. verbundenen Kosten ebenfalls kurzfristig mitteilen. Bei der Antwort wird der Auftragnehmer die Auswirkungen des Änderungswunsches auf Termine und die vollständigen Kosten für das Durchführen der Änderung mitteilen.“

Bei MiN lautet die entsprechende Formulierung

„7.3 Der Projektleiter des AG ist für die Koordination und Überwachung aller Änderungs- und Erweiterungs- oder Kürzungswünsche bzgl. der im Pflichtenheft und/oder dem Abweichungsbericht niedergelegten Vereinbarungen (Change-Management) verantwortlich. Der AN wird Anfragen aus dem Change-Management des AG kurzfristig beantworten. Macht ein Änderungsverlangen eine umfangreichere Prüfung erforderlich, wird der AN dies und die mit der Prüfung ggf. verbundenen Kosten ebenfalls kurzfristig mitteilen. Bei der Antwort wird der AN die Auswirkungen des Änderungswunsches auf Kosten und ggf. auf Termine berücksichtigen.“

Dies wird bei MiN unter den Bestimmungen zur Wartung (**13. Wartung**) ergänzt.

„13.4 Updates und Upgrades

Gegenstand des Wartungsvertrages ist ferner die kostenlose Lieferung neuer, verbesserter und erweiterter Programmversionen (Updates und Upgrades) der Software. Dabei stellt der AN sicher, dass die vorhandenen Daten übernommen werden und auch die individuell angepassten Programme mit den neuen Versionen zusammenarbeiten. Ggf. sind im Rahmen des Wartungsvertrages entsprechende Anpassungsleistungen zu erbringen. Der AN verpflichtet sich, gesetzlichen Änderungen mit für den AN erkennbaren Auswirkungen auf die Funktionalität der Software beim AG rechtzeitig vor Wirksamwerden der Änderung im Rahmen des Wartungsvertrages zu liefern und sichert die Jahr 2000 Fähigkeit der Software ausdrücklich zu.

Der AN liefert die Updates in regelmäßigen Abständen und gewährleistet einen einwandfreien Änderungsdienst mit eindeutiger Versionsverwaltung.“

Es findet sich das bereits bekannte dreistufige Schema wieder, wonach einer Anfrage des Softwarebestellers eine Prüfung des Softwarehauses folgt. Das Ergebnis der Prüfung muss Angaben zu Kosten und Terminen enthalten. Abgeschlossen wird das Verfahren mit der, hier nicht gesondert aufgeführten, Entscheidung des Softwarebestellers. Es kommt hier eine Zustimmung oder eine Ablehnung in Betracht, eine Regelung kontroverser Auffassungen zu den Änderungskonditionen besteht nicht.

Aus der Wartungsregelung bei MiN wird deutlich, dass die Vertragspartner von einer beständigen Änderung („in regelmäßigen Abständen“ gelieferte Updates – was laut Vertragsdefinition in 13.4 S. 1 verbesserten Programmversionen entspricht) ausgehen. Dies erklärt, warum bei MiN den Änderungsregelungen keine große Beachtung geschenkt wird.

Bei MiO werden die Änderungen unter „**4. Änderungen der Vorgaben für Anpassungen**“ behandelt:

„Geringfügige Änderungen einer Vorgabe (Zielvorgabe, Vorstudie, Pflichtenheft, einzelne Funktionen) werden durch Absprache zwischen dem Projektleiter des Auftraggebers und dem Projektleiter des Auftragnehmers vorläufig festgelegt, an jedem Monatsende vom Projektleiter des Auftragnehmers schriftlich festgehalten und werden durch kaufmännischen Briefwechsel zwischen den Parteien vereinbart. Auftraggeber und Auftragnehmer bevollmächtigen hiermit in diesem Sinne ihre Projektleiter.

Größere Änderungen bedürfen von vorne herein der Schriftform.

Als größere Änderungen sind Änderungen zu verstehen, deren Implementation entweder den Arbeitsumfang von drei Manntagen überschreitet oder die Funktionalität der Software einschränkt.

Kleinere Änderungen bis zum Gesamtaufwand von sechs Arbeitstagen je Quartal des Projektzeitraumes sind im Pauschalpreis inbegriffen. Rechtzeitig vor der Inangriffnahme der Ausführung weiterer Änderungen muss der Auftragnehmer die schriftliche Zustimmung des Auftraggebers einholen.“

Es wird, terminologisch nicht ganz klar, zwischen kleinen (geringfügigen) und größeren Änderungen unterschieden. Dem in Absatz 1 festgelegten monatlichen Rhythmus ist die Üblichkeit der Änderungen zu entnehmen. Es wird kein besonderes Verfahren vereinbart, sondern von einer gemeinsamen Entscheidung ausgegangen.

Auch bei MiO ist im Rahmen der Wartung die Lieferung von neuen Versionen vorgesehen.

„6. Vertragsgegenstand Wartung

(...)

Updatewartung i.S. dieses Vertrages bedeutet die Lieferung von neuen Systemversionen (Updates, Releases) inklusive Sourcecode, wie sie standardmäßig allen anderen Kunden des Auftragnehmers angeboten bzw. geliefert werden für die eingesetzten Programmmodule. Updatewartung garantiert die laufende Anpassung der Anwendungssoftware an notwendige Änderungen der Umgebungsbedingungen wie Hardware, Systemsoftware inklusive Datenbanksysteme, Steuerrecht in Österreich, Deutschland, Schweiz und der EU.“

Interessant ist die ebenfalls unter 6. definierte „**Basiswartung**“:

„Basiswartung im Sinne dieses Vertrages bedeutet:

a) Die Unterstützung bei der Lokalisierung von Mängeln und die Beseitigung von Störungen und Mängeln. Die Beseitigung von Störungen und Mängeln erfolgt durch Zurverfügungstellung einer mangelfreien Programmversion per Datenträger oder Datenübertragung.“

Es wird hier erneut deutlich, dass die Betreuung über den gesamten Lebenszyklus gefordert ist. Die Basiswartung ist gem. der Vergütungsvereinbarung in Nr. 12 Abs. 4 „bis zum Ende der Gewährleistungsfrist (9 Monate ab Gesamtannahme)“ im Fixpreis inkludiert. Danach wird sie für die folgenden 15 Monaten gegen monatliche Vergütung erbracht. Die Parteien gehen also offenbar davon aus, dass auch nach der (End-) Abnahme „Störungen und Mängel“ zu beseitigen sind, und der Softwarebesteller ist bereit, dafür eine regelmäßige Vergütung zu leisten.

VII. Sonstiges

Aus den Gewährleistungsregeln ergeben sich keine besonderen Leistungsmaßstäbe. MiO und MiN halten sich sehr kurz:

„Treten gewährleistungspflichtige Mängel auf, wird der Auftragnehmer diese beheben (...)“ (MiO Nr. 20 Abs. 4).

„Die Gewährleistung des AN richtet sich grundsätzlich nach §§ 631ff. BGB“ (MiN Nr. 8.1).

Etwas ausführlicher ist die Regelung in SüS unter „**11. Gewährleistung**“:

„(...)

11.2 Als Mängel gelten Abweichungen der Vertragssoftware von der im Pflichtenheft beschriebenen Funktionsweise, soweit diese Abweichungen die Tauglichkeit zum üblichen, im Pflichtenheft beschriebenen Funktionsweise beeinträchtigen. Die Gewährleistung besteht nicht, wenn der Mangel nur unerheblich ist, sich also insbesondere nicht erheblich auf die Gebrauchstauglichkeit auswirkt.“

Während MiN auf den gesetzlichen Maßstab verweist, nimmt SüS ausdrücklich auf das Pflichtenheft Bezug. Im Ergebnis dürfte dies allerdings zu keinem Unterschied führen, da das Pflichtenheft zusammen mit dem Abweichungsbericht die zwischen den Parteien vereinbarte Gebrauchseignung bestimmt.

Alle Verträge enthalten Fixpreisregelungen, wobei in MiN und MiO quartalsweise bzw. monatliche Zahlungen für die Wartung vorgesehen sind.

C. Zusammenfassung und Ergebnis

Die Autoren der Muster sind überwiegend ersichtlich darum bemüht, einen eigenen Ansatz zu verfolgen. Dies zeigt sich am deutlichsten bei der unterschiedlichen Terminologie etwa hinsichtlich des Vertrages selbst, dem Liefergegenstand oder der Dokumentation, aber auch beim Vertragsaufbau, den Fragen von Sourcecodeübergabe und dem Umfang der Übertragung von Nutzungsrechten. Dabei neigen die Muster auch zu einer abstrakten Formulierungsweise. Es wird ganz überwiegend versucht, einen Hauptvertrag mit abschließender juristischer Regelung, ergänzt um Anlagen zur Klärung der technischen Fragen, zu konzipieren.

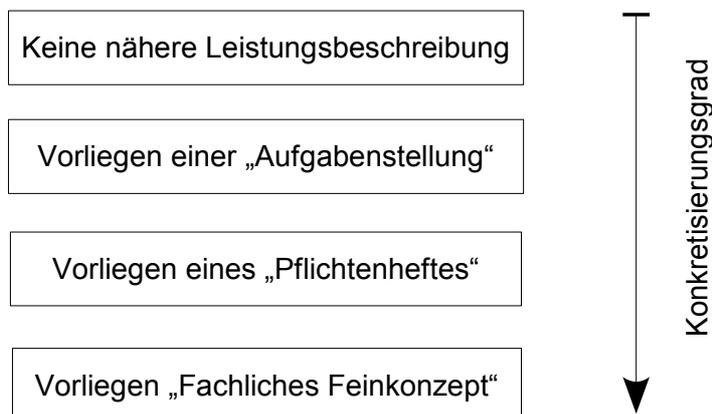
Eine Vergleichbarkeit der Verträge ist nach dem Vorstehenden nur eingeschränkt möglich.

In den Praxisbeispielen wird bereits durch die Benennung des Vertrages in zwei Fällen eine klare Einordnung als Werkvertrag versucht. Dabei werden eine Vielzahl auch technischer Details bereits im Hauptvertrag geregelt.

Muster und Praxisbeispiele sind - von verschiedenen Ausnahmen abgesehen - um Ausgewogenheit bemüht und bürden keinem der Vertragspartner übermäßige Pflichten auf. Tendenziell begünstigen die Muster und Praxisbeispiele eher, von einer Ausnahme abgesehen, in nachvollziehbarem Maße den Softwarebesteller.

In fast allen Mustern fehlt es an einer (hypothetischen) Einführung in die tatsächlichen Gegebenheiten.

Eine unterschiedliche Ausgangssituation wird nicht erklärt, sondern vorausgesetzt. Für den Leser (und Verwender) wird häufig erst bei genauem Studium aus dem Zusammenhang ersichtlich, in welcher Phase sich das Projekt bei Vertragsschluss tatsächlich befindet. Das Spektrum des Projektstandes reicht in den Mustern von „Null“, d.h. die Parteien treffen sich auf Basis einer mehr oder weniger konkretisierten Idee seitens des Softwarebestellers, bis hin zu dem Vorliegen einer ausgefeilten Formulierung (fachlichen Feinspezifikation) oder einem bereits geprüften Pflichtenheft.



Da die Begriffe nicht einheitlich verwendet werden, kann das Pflichtenheft ähnlich konkrete oder sogar detailliertere Angaben als das Feinkonzept bzw. die Feinspezifikation enthalten.

Wird von der Vertragsabwicklung auf Basis eines Pflichtenheftes ausgegangen, so ist dieses teilweise vom Softwarebesteller, teilweise vom

Softwarehaus und teilweise gemeinsam zu erstellen, in allen Praxisbeispielen wurde es vom Softwarebesteller vorgelegt. Hat der Softwarebesteller die Aufgabe der Erstellung des Pflichtenhefts übernommen, ist in den Mustern eine korrespondierende Pflicht von der Beratung bis hin zur Prüfung vorgesehen. Übernimmt sie das Softwarehaus, so ist teilweise eine Abnahme durch den Softwarebesteller gefordert. Die Terminologie ist in den Praxisbeispielen nicht eindeutig.

Eine im Projektverlauf zunehmende Konkretisierung ist in einem Muster und - eingeschränkt - aus den Praxisbeispielen ablesbar, wobei sie sich in einem Praxisbeispiel bereits vor dem Vertragsschluss vollzogen hat. Das Pflichtenheft wurde jedenfalls stets vom Softwarehaus geprüft und dient in zwei Fällen zusammen mit anderen, vorvertraglichen Unterlagen als Grundlage der Leistungsbestimmung. In dem Praxisbeispiel, wo die Anforderungen im Vorfeld präziser festgelegt werden konnten und eine Integration in eine bestimmte betriebliche (Soft- und Hardware-) Umgebung nicht erforderlich ist, findet sich eine deutlich geringere Regelungsdichte.

Die besonderen Pflichten der Vertragspartner dienen ganz überwiegend zur Projektkontrolle. So ist das Softwarehaus gehalten, in inhaltlich unterschiedlich ausgestaltetem Maße, regelmäßig über den Projektstand zu informieren. Personalfragen wird keine hohe Bedeutung beigemessen, teilweise werden Mitarbeiterqualifikationen gefordert, in einem Beispiel behält sich der Softwarebesteller in Ausnahmesituationen einen Einfluss auf Personalentscheidungen vor.

Die Pflichten des Softwarebestellers beschränken sich nahezu durchgehend auf das Bereitstellen von Ressourcen, zum Teil in rudimentärer Form, und auch dies wird durch weitere Einschränkungen nicht unerheblich relativiert. Eine aktive Rolle des Softwarehauses ist mit zwei Ausnahmen, wobei eine insoweit nur beschränkte Regelungen enthält, nicht vorgesehen. Dies gilt auch für die Praxisbeispiele, wobei die Tendenz, Mitwirkungspflichten zu Gunsten des Softwarebestellers einzuschränken, besonders bei dem Vertrag zur Erstellung von Standardsoftware auffällt.

Die Notwendigkeit für die Regelung von Veränderungen wird teilweise gering eingeschätzt, und der Vertragsinhalt fällt dementsprechend kurz aus. In einigen Mustern wird in der Änderungsregelung im Gegensatz dazu ein Schwerpunkt des Vertrages gesehen. Die Initiative für Veränderungen soll dabei ganz überwiegend beim Softwarebesteller liegen. Es finden sich mehrstufige Verfahren mit zum Teil ausführlichen Detailregelungen. Dabei zeigt sich ein ausgeprägtes Bedürfnis, auch Änderungen formal zu erfassen. In zwei Praxisbeispielen werden Änderungen als selbstverständlich vorausgesetzt, sollen jedoch nicht in einem umfangreichen Verfahren abgehandelt, sondern v.a. im Rahmen der Wartung durchgeführt werden. Insgesamt zeichnet sich der Wunsch des Softwarebestellers nach einer dauerhaften, über die Softwareentwicklung hinausgehenden Betreuung ab (in den Mustern sind Pflege und Wartung entweder nicht vorgesehen oder Gegenstand eines getrennt behandelten Entwurfes). Besonderer Wert wird also auf die dauerhafte Sicherung des Auftragszieles gelegt.

Aus dem Gewährleistungsmaßstab lässt sich wenig für die Leistungsbestimmung entnehmen. Die teilweise vorhandenen Verweise auf technische Standards sind von zweifelhaftem praktischen Nutzen. Die Fehlerbeseitigung folgt, von einer Ausnahme abgesehen, stets der Abnahme und wird als Gewährleistungsfrage behandelt. In den Praxisbeispielen wird sie als Bestandteil der Wartung betrachtet

Die Frage nach der Projektleitung ist nicht in allen Mustern, und dort, wo sie vorhanden ist, unterschiedlich geregelt. Sie wird dem Softwarehaus, aber auch dem Softwarebesteller oder einem paritätisch besetzten Gremium übertragen.

Sowohl die Muster als auch die Praxisbeispiele enthalten ohne Ausnahme Festpreisvereinbarungen.

4. RECHTSVERGLEICHUNG

Kapitel 7: Vergleichende Darstellung – Methoden der Leistungsbestimmung

Weder im deutschen noch im amerikanischen Recht ist die Softwareentwicklung durch Gesetze bzw. statutes geregelt. Im amerikanischen Recht ist der Versuch, ein auch die Softwareentwicklung regelndes Uniform Law auf den Weg zu bringen, nach anfänglichen Erfolgen gescheitert. Im deutschen Recht gibt es zwar Grundmodelle wie den Werkvertrag, die als Regelungsbasis in Betracht kommen. Welches Modell sich jedoch in welchem Ausmaß eignet, ist heftig umstritten und infolge der Schuldrechtsreform auch aus Sicht der Rechtsprechung fraglich.

Die Notwendigkeit, Software in das hergebrachte Rechtssystem einzuordnen, trifft in beiden Rechtsordnungen auf erhebliche Schwierigkeiten. Im amerikanischen Recht führt dies dazu, dass z.B. die Frage, ob ein Gericht den U.C.C. anwenden wird oder nicht, keineswegs sicher beantwortet werden kann. Dieser Befund ist auch das Ergebnis der unterschiedlichen Gerichtsbarkeiten und Rechtsgrundlagen in den 50 amerikanischen Bundesstaaten.

In der deutschen Rechtssphäre zeigt sich die Problematik auf andere Weise. Während die Gerichte überwiegend einer klaren Linie folgen, findet sich in der Fachliteratur eine kaum überschaubare Vielfalt von Meinungen. In den USA würde eine wie in Deutschland gefestigte Rechtsprechung die Rechtslage mit hoher Wahrscheinlichkeit über viele Jahre hinweg zementieren. In Deutschland, wo es eine Bindung der Gerichte an Vorentscheidungen nicht gibt, kann sich die Rechtslage durch eine Maßnahme des Gesetzgebers schnell ändern. Durch die Schuldrechtsreform aus dem Jahre 2002 und vor allem die damit verbundene Änderung des nach der Rechtsprechung maßgeblichen Werkvertragsrechts ist genau eine solche Situation eingetreten. Die deutschen Gerichte könnten diese Gesetzesänderung zum Anlass nehmen, ihre Entscheidungspraxis zu

ändern. Sie sind dazu aber nicht gezwungen, weil das Gesetz (nur) das bisher als Grundlage der Spruchpraxis dienende Werkvertragsrecht änderte, nicht aber die Softwareentwicklung oder sonstige Softwareverträge regelte⁵³⁰. Eine Maßnahme der Legislative kann die Rechtsprechung also zwingen bzw. bezüglich des Untersuchungsgegenstandes ihr die Möglichkeit eröffnen, neue Wege zu gehen, und dies im gesamten Bundesgebiet und für alle Gerichte⁵³¹.

Bisher ist eine Trendwende – soweit ersichtlich – nicht erkennbar. Die Entwicklung ist jedoch offen, d.h. wie in den USA kann eine zuverlässige Prognose über die Rechtsgrundlagen, die einer gerichtliche Entscheidung als Basis dienen werden, nicht getroffen werden. Während der Grund hier in einer Gesetzesänderung, die neue Subsumtionsmöglichkeiten eröffnet, zu suchen ist, liegt er dort in der bisher uneinheitlichen Entscheidungspraxis.

Die Grundvoraussetzungen für die Softwareentwicklung aus informationstechnologischer Sicht wurden im Kapitel 2 ausführlich dargelegt. Für dieses Kapitel sind zunächst die dort gewonnenen Erkenntnisse fruchtbar zu machen. Dies geschieht im Folgenden anhand der Ausgangsfrage, ob und ggf. wie die Verträge und Muster die Vorgaben des Software-Engineering berücksichtigen. Im Einzelnen:

- 1) Wird ein präzises Ziel vorgegeben und wie geschieht dies ggf.?
- 2) Ist ein Reaktionsmechanismus für einen Wandel der Anforderungen vorgesehen und ggf. in welcher Form?
- 3) Wird die Mitarbeit des Softwarebestellers sichergestellt und ggf. auf welche Weise?
- 4) Werden angemessene Qualitätskriterien für die Beurteilung der Leistung gewählt, und wird dabei insbesondere die mangelnde Eignung der Fehlerfreiheit berücksichtigt?

⁵³⁰ Vor diesem Hintergrund werden die zahlreichen Stellungnahmen aus der rechtlichen Literatur verständlich, da hier eine Gelegenheit gesehen wird, auf die weitere Entwicklung Einfluss zu nehmen. Bei unklarer Rechtslage bedienen sich die Gerichte gerne der Stellungnahmen aus der Literatur, wenn es an (ober-)gerichtlichen Entscheidungen fehlt.

⁵³¹ Dieser letzte Hinweis wäre in einer auf das deutsche Recht beschränkten Darstellung trivial, für einen amerikanischen Kollegen dagegen kann er aus den in Kapitel 3 dargelegten Gründen durchaus überraschend sein.

Dabei sind die gesetzlichen Rahmenbedingungen daraufhin zu untersuchen, ob sie diesbezügliche Vorgaben und Einschränkungen enthalten.

A. Zielpräzision

Aus inhaltlicher Sicht erweist sich das *Statute Law* als wenig ergiebig. Selbst bei Anwendung des U.C.C. lassen sich diesem allenfalls Anhaltspunkte generellen Inhalts entnehmen. Es obliegt den Parteien, den Inhalt der gegenseitigen Pflichten durch einen Vertrag zu bestimmen. Ein Rückgriff auf vorvertragliche Vereinbarungen zur Definition oder Auslegung dieser Pflichten wird als *extrinsic evidence* grundsätzlich ausgeschlossen. Tatsächliche Vertragsdurchführung, zwischen den Parteien übliche Vertragsgestaltungen und Handelsbräuche können jedoch für die Erklärung und Ergänzung der Vereinbarungen verwendet werden. Im Ergebnis verbleibt es bei einem sehr weit gesteckten Rahmen.

Eine Besonderheit genereller Art kann aus den *Restatements* für Fälle abgeleitet werden, in denen ein Gericht die Parteivereinbarung für unvollständig oder für *unconscionable*⁵³² hält. Es steht ihm dann frei, eine den Umständen angemessene Regel hinzuzufügen, bzw. den Vertrag insgesamt oder bezüglich bestimmter Teile für unwirksam zu erklären. Dies gäbe theoretisch Raum dafür, auch hinsichtlich der Zielpräzision Ergänzungen bzw. Subtraktionen vorzunehmen, eine Möglichkeit, von der in dem hier untersuchten Bereich allerdings kein Gebrauch gemacht wurde.

Das Case Law hat sich in der in rechtlichen Dimensionen insbesondere im Common Law verhältnismäßig kurzen Zeitspanne von zwanzig Jahren als dynamisch erwiesen. Wurde es noch 1984 für selbstverständlich gehalten, dass das Softwarehaus für den Spezifikationsvorgang weitgehende Verantwortung übernimmt, so zeigt sich in neueren Entscheidungen, dass der Softwarebesteller zunehmend mehr am Projektrisiko beteiligt wird.

Bei zusammenfassender Betrachtung lassen sich im amerikanischen Recht keine Vorgaben für die Präzision des Vertragszieles erkennen. Tendenziell

⁵³² Zum Begriff siehe oben, Kapitel 3 S. 52.

ist eine einseitige Belastung eines Vertragsbeteiligten mit der Verantwortung für detailgenaue Spezifikationen nicht unproblematisch. Bei ausdrücklicher schriftlicher Vereinbarung und nicht völlig ungleicher Verhandlungsposition sind den vertraglichen Regelungen diesbezüglich allerdings keine Grenzen gesetzt.

Dem BGB als der maßgeblichen deutschen Rechtsquelle lässt sich kein klares Bild für die Frage nach Vorgaben bzw. Einschränkungen bei der Leistungspräzision entnehmen. Zunächst ist unklar, welches Modell (vor allem Kauf-, Werk- oder modifiziertes Werkvertragsrecht) zu Grunde zu legen ist. Doch selbst wenn ein Vertragstypus gewählt ist, so finden sich keine näheren Regeln zum Umfang der Leistungsbestimmung. Sowohl der Vertragszweck „Herstellung des versprochenen Werkes“ als auch Übergabe und Eigentumsverschaffung überlassen es den Vertragsparteien, eine genauere Bestimmung der Pflichten vorzunehmen.

In deutscher Literatur und Rechtsprechung wird das Thema der Leistungspräzisierung meist unter dem Stichwort „Pflichtenheft“ abgehandelt. Einigkeit besteht dabei über ihre Notwendigkeit. Gegenstand von Gerichtsentscheidungen werden jedoch vor allem Fälle, in denen ein Pflichtenheft nicht erstellt wurde. Manche Softwarehäuser setzen sich danach über die in ihrer Branche entwickelten Standards hinweg und entwickeln Software ohne Zielpräzision. Die Rechtsprechung sieht die Verantwortung für das Erstellen des Pflichtenhefts ganz überwiegend beim Softwarebesteller. Dabei darf allerdings nicht unbeachtet bleiben, dass unter einem Pflichtenheft nicht in jedem Fall eine detaillierte Leistungsvorgabe verstanden wird. Wenn Literatur und Rechtsprechung die vorgenannte Aufgabenzuweisung zum Softwarebesteller vornehmen, so ist damit häufig (nur) eine konkrete Darstellung der programmtechnisch zu lösenden Aufgabe gemeint. Dies ist aus dem Blickwinkel der Informationstechnologie zwar mehr als nur eine Idee, aber weitaus weniger als eine präzise Grundlage für die Implementation. Ist eine solche Grundlage von Seiten des Softwarebestellers geschaffen, so wechselt nach der Rechtsprechung die Verantwortung für den weiteren Projektverlauf grundsätzlich zum Softwarehaus.

Weder im amerikanischen noch im deutschen Recht liefert das in den Gesetzen enthaltene Recht nach dem Vorstehenden Aufschlüsse über die Zielpräzision. Ausgehend von allgemeinen Rechtsgrundsätzen wird die Verantwortlichkeit für die Zielpräzision in der Rechtsprechung keinem Vertragspartner eindeutig zugewiesen, wobei sich, über einen Zeitraum von etwa 20 Jahren gesehen, eine Tendenz dazu zeigt, den Softwarebesteller mehr in die Pflicht zu nehmen.

Die gesetzlichen Rahmenbedingungen enthalten für eine detaillierte Leistungsbestimmung kaum Vorgaben oder Einschränkungen. Die Vertragsgestaltung ist daher weitgehend frei möglich. Dies kommt in den Vertragsmustern und -beispielen auch zum Ausdruck.

Der überwiegende Teil der amerikanischen Muster und zwei der Praxisbeispiele gehen von einem Vertragsabschluss in einem frühen Projektstadium aus, d.h. von einem Zeitpunkt, zu dem der Detaillierungsgrad der Anforderungen noch vergleichsweise gering ist, ein Großteil der Spezifikationsarbeit also noch vor den Parteien liegt.

Es finden sich durchgehend mehrstufige Verfahren zur Bestimmung des Vertragsinhaltes. Sie werden durch eine Anforderungsformulierung eingeleitet, der sich die Erstellung eines präzisen Anforderungsprofils anschließt und die schließlich in die Implementation, d.h. programmtechnische Umsetzung, mündet. Dabei sind auch technische Details bereits im Hauptvertrag vorhanden. Die Erstellung der Spezifikation auf diesem Wege wird teilweise für die Zukunft vorgesehen. In zwei Praxisbeispielen wird die Spezifikation ebenfalls sukzessive im Projektverlauf vorgenommen, im dritten ist sie fest vorgegeben. Die Erstellung technischer Spezifikationen wird in den Mustern ganz überwiegend als Sache des Softwarehauses angesehen, in zwei Praxisbeispielen wird sie als gemeinsame Aufgabe der Vertragsparteien betrachtet, in einem wurde sie vom Softwarehaus erstellt und vom Softwarebesteller geprüft.

Im Ergebnis wird die Erarbeitung detaillierter Anforderungen als Ziel vorgegeben, was durch die Vorgehensweise in einem abgestuften Verfahren

erreicht werden soll. Die Verantwortlichkeit ist dabei in einigen Mustern nicht eindeutig zugewiesen, eine Tendenz, die auch in zwei Praxisbeispielen bestätigt wird und das Element der gemeinsamen Entwicklung betont. Aus informationstechnischer Sicht wird dem Erfordernis nach Zielpräzision weitgehend Rechnung getragen.

Die deutschen Muster basieren auf unterschiedlichen Stadien der Softwareentwicklung, wobei auch hier ein mehrstufiges Verfahren zur Leistungsbestimmung in der Mehrzahl der Muster vorgesehen ist. Es wird teilweise davon ausgegangen, es läge ein – geprüftes - Pflichtenheft vor, welches nach Maßgabe des Vertrages umzusetzen ist.

Auch zwei der Praxisbeispiele gehen von einer mehrstufigen Detaillierung der Anforderungen aus, die sich in einem Beispiel während des Projektverlaufes fortsetzt. In zwei Beispielen werden die Anforderungen bei Vertragsabschluss als vollständig betrachtet. Die Verantwortlichkeit für das Erstellen eines Pflichtenhefts, als zentrales Spezifikationsdokument, ist unterschiedlich geregelt und teilweise als Option vorgesehen.

Technische Details werden weitgehend in die Anlagen verwiesen.

Auch die deutschen Muster und Praxisbeispiele bemühen sich darum, eine möglichst detaillierte Leistungsbestimmung durch ein mehrstufiges Verfahren für deren Erarbeitung sicherzustellen. Wie bei ihren amerikanischen Pendanten ist die Verantwortung dabei unterschiedlich geregelt. Die Vorgabe der Zielpräzision wird damit weitgehend ausreichend berücksichtigt. Dies geschieht auf vergleichbare Weise. Die amerikanischen Muster und ihre deutschen „Kollegen“ bemühen sich mit einer Ausnahme darum, den „juristischen“ (Haupt-)Vertrag von der in Anlagen zu regelnden Technik zu trennen⁵³³. Die Praxisbeispiele aus beiden Gruppen nehmen auch technische Details in den Hauptvertrag auf.

⁵³³ Dies versteht sich einerseits, da es sich um für eine Vielzahl von Fällen vorgesehene Muster handelt, von selbst. Andererseits wäre es durchaus, gerade im Bereich der Beschreibung der Vorgehensweise bei der Leistungsbestimmung, durchaus möglich, diese im Hauptvertrag zu verankern. So ist es beispielsweise bei HE vorgesehen, der allerdings einen sehr technisch-abstrakten Ansatz verfolgt und die konkreten Fragen im Projektleben nicht regelt.

B. Reaktionsmechanismen bei Änderungen

Das amerikanische und das deutsche Recht enthalten Regeln für den Fall von Änderungen der Rahmenbedingungen.

So heißt es in den *Restatements 2d § 89 a*, dass (auch einseitige) Änderungen der in einem Vertrag begründeten Pflichten verbindlich sind, wenn sie sich als „*fair and equitable in view of circumstances not anticipated when the contract was made*“ darstellen⁵³⁴. Regelungsgegenstand dieser Norm ist die Rechtslage bei *unvorhergesehenen* Ereignissen⁵³⁵.

Im deutschen Recht wird Änderungen nach Vertragsabschluss nunmehr in § 313 BGB unter dem Titel „Störung der Geschäftsgrundlage“ Rechnung getragen. Dort heißt es:

(1) Haben sich Umstände, die zur Grundlage des Vertrags geworden sind, nach Vertragsschluss schwerwiegend verändert und hätten die Parteien den Vertrag nicht oder mit anderem Inhalt geschlossen, wenn sie diese Veränderung vorausgesehen hätten, so kann Anpassung des Vertrags verlangt werden, soweit einem Teil unter Berücksichtigung aller Umstände des Einzelfalls, insbesondere der vertraglichen oder gesetzlichen Risikoverteilung, das Festhalten am unveränderten Vertrag nicht zugemutet werden kann.

(2) Einer Veränderung der Umstände steht es gleich, wenn wesentliche Vorstellungen, die zur Grundlage des Vertrags geworden sind, sich als falsch herausstellen.

Auch hier ist von einer unvorhergesehen Änderung der Umstände die Rede, die zudem schwerwiegend sein muss.

⁵³⁴ Dass nicht jede von den Parteien gewollte Vertragsänderung verbindlich ist, ergibt sich aus dem Erfordernis der sog. *consideration*, ein dem deutschen Recht unbekanntes Prinzip. Vereinbarungen sind danach nur dann bindend, wenn ihnen, grob gesprochen, ein Austausch, d.h. ein Geben und Nehmen zugrunde liegt. Fehlt es daran, wie im Fall von Geschenken, so liegt keine einklagbare Schuld vor, vgl. Hay, Peter, U.S.-Amerikanisches Recht, 2. Aufl., München, 2002, Rn. 298ff; Calamari/Perillo, a.a.O., 4.1, S. 168ff und 5.14, S. 237ff.

⁵³⁵ Dasselbe gilt für die Parallelnorm im Handelsrecht, § 2-209 U.C.C.

Änderungen der Anforderungen in Softwareentwicklungsverträgen sind nach den Erfahrungen der Praxis, wie sie sich auch in zahllosen Entscheidungen der Gerichte widerspiegeln, vorhersehbar. Die Vertragsparteien wissen in aller Regel von dieser Notwendigkeit und berücksichtigen sie ebenso regelmäßig in den Verträgen⁵³⁶.

Für die rechtlichen Rahmenbedingungen für Änderungsmechanismen gilt hier also das bereits zur Zielpräzision Ausgeführte, d.h. es obliegt den Vertragsparteien ohne nähere Vorgaben oder Einschränkungen des Rechts, wie sie Änderungen der Anforderungen im Projektverlauf regeln möchten.

In vier der amerikanischen und sieben der deutschen Muster sowie in zwei amerikanischen und allen deutschen Praxisbeispielen (bei letzteren in zwei davon im Rahmen von Pflegevereinbarungen) ist ein Change Management vorgesehen.

In den an den Interessen des Softwarehauses orientierten amerikanischen Mustern ist es nicht geregelt, wohl dagegen im entsprechenden deutschen Muster.

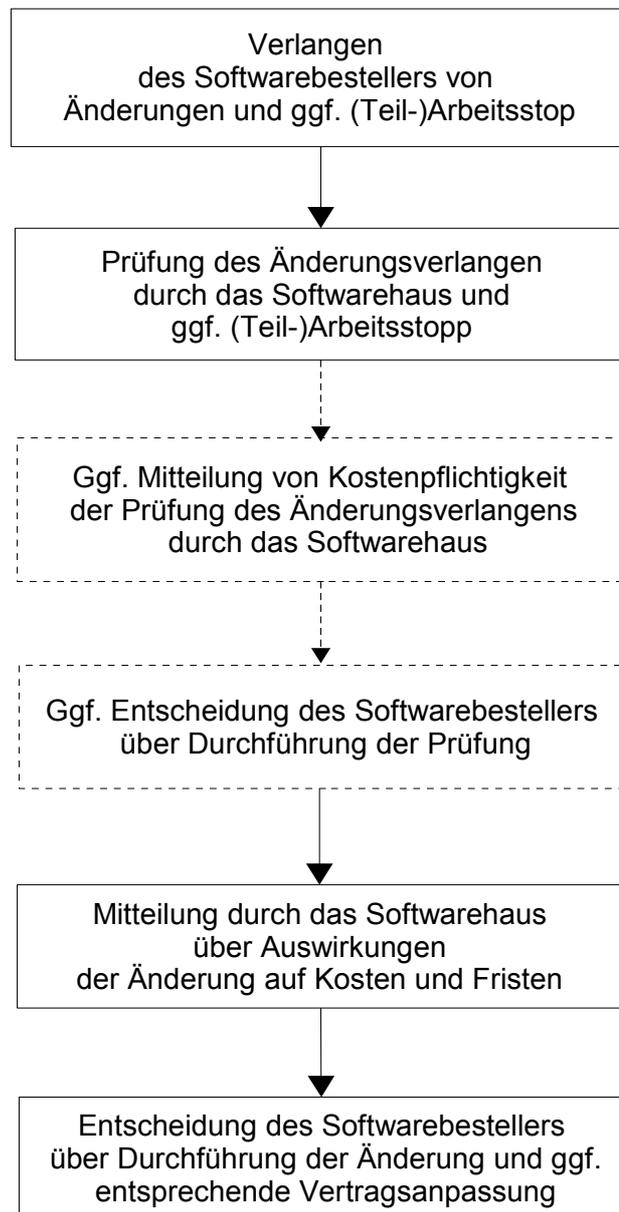
Der Inhalt der jeweiligen Regelung fällt sehr unterschiedlich aus. Teilweise wird zwischen kleinen und größeren Änderungen unterschieden.

Ausgangspunkt ist ein Änderungsverlangen, das in jeweils einem deutschen und amerikanischen Muster sowie einem amerikanischen Praxisbeispiel auch vom Softwarehaus ausgehen kann. Ansonsten liegt die Initiative stets beim Softwarebesteller.

Die Durchführung kann in einem amerikanischen Praxisbeispiel und in zwei Mustern vom Softwarebesteller verlangt und in zwei Mustern vom Softwarehaus auch abgelehnt werden. Unter den teilweise sehr ausführlichen deutschen Mustern finden sich zwei, in denen ein Recht zur Ablehnung von Änderungen vorgesehen, und vier, in denen es ausgeschlossen ist, wobei eines davon die Option der Ablehnung beinhaltet.

⁵³⁶ Dies schließt nicht aus, dass sich im Einzelfall die Umstände dergestalt ändern, dass eine Anpassung nach den genannten Vorschriften verlangt werden kann. Es handelt sich dann jedoch nicht um eine Frage der Leistungsbestimmung, sondern einen allgemeinen Grundsatz, der nicht Gegenstand dieser Untersuchung ist.

Das Verfahren für Änderungen folgt ganz überwiegend diesem zeitlichen Schema:



Eine vergleichende Übersicht zu den Änderungsmechanismen ist der folgenden Tabelle zu entnehmen.

Merkmal des Änderungsverfahrens*	Deutsch		US	
	Muster	Beispiele	Muster	Beispiele
Vorhanden	7	3	4	2
Unterscheidung zwischen großen und kleinen Änderungen	1	1	1	
Einleitung nur durch Softwarebesteller	6		2	
Einleitung auch durch Softwarehaus**	1		1	1
Änderung kann vom Softwarebesteller verlangt werden***	4	1	2	1
Änderungswunsch des Softwarebestellers kann abgelehnt werden	2	1	2	
Das Verfahren ist an Fristen gebunden	3		2	

* Es handelt sich überwiegend nicht um klare Ja/Nein Kriterien, weshalb die Tabelle nur als Übersicht zur Verdeutlichung einer Tendenz gedacht ist. Auch muss natürlich die (geringe) Größe der Stichprobe hinsichtlich der Praxisbeispiele berücksichtigt werden. Zur Erinnerung: Es wurden 9 deutsche und 7 amerikanische Muster und jeweils 3 Praxisbeispiele untersucht.

** Dass dies in den Beispielen nicht ausdrücklich vorgesehen ist, liegt teilweise an der Vertragsprämisse, Änderungen würden durch die von beiden Vertragsparteien besetzte Projektleitung kontinuierlich abgehandelt

***Diesem Verlangen muss in drei Fällen bei Unzumutbarkeit nicht entsprochen werden. Auch wenn eine Änderung verlangt werden kann, setzt sie bei größerem Ausmaß in jedem Fall eine Vereinbarung über die Vergütung voraus.

Die Bewertung dieser Informationen gestaltet sich infolge der unterschiedlichen Ausgangspunkte der Muster und Beispiele nicht einfach. Es lassen sich jedoch im Hinblick auf die Ausgangsfrage, ob und ggf. wie einem Wandel der Anforderungen Rechnung getragen wird, einige klare Aussagen treffen.

Auch wenn in elf Mustern und fünf Beispielen ein Change-Management vorgesehen ist, so überrascht hinsichtlich der Muster, dass es bei fünf von ihnen, also nahezu einem Drittel, an einer Regelung fehlt. Dort, wo sie vorhanden ist, kann ein Änderungsverlangen, von jeweils einer Ausnahme abgesehen, nur von dem Softwarebesteller ausgehen. Die Möglichkeit der Ablehnung durch das Softwarehaus ist (nur) in jeweils 2 Mustern und einem Praxisbeispiel vorgesehen.

Zusammengefasst ergibt sich folgendes Bild. Die Regelung des Change Managements wird von einer bedeutenden Anzahl von Mustern vernachlässigt. Dass Änderungsverfahren fast durchweg nicht vom Softwarehaus eingeleitet und zudem nur ausnahmsweise von diesem zurückgewiesen werden können, ist nur als Entgegenkommen seitens des Softwarehauses verständlich und möglicherweise Ausdruck der derzeitigen Machtverhältnisse zwischen Softwarebestellern und Softwarehäusern⁵³⁷. Aus Sicht des Software Engineering ist es nicht nachvollziehbar und stellt eine signifikante Gefahr für den Projekterfolg, d.h. die termin- und budgetgerechte Erfüllung, dar. Dieses Problem wird, mit wenigen Ausnahmen, weder von deutschen noch von amerikanischen Autoren thematisiert. Der Vergleich zeigt also eine Fehleinschätzung im Verhältnis der Disziplinen und nicht der Rechtsordnungen zueinander⁵³⁸. Soweit Unterschiede ersichtlich sind, bestehen sie zum einen in der ausgeprägteren Tendenz deutscher Muster, Technik und Recht trennen zu wollen. Zum anderen äußern sie sich in deren Wunsch, Änderungen „im Griff“ zu haben, d.h. sich durch detaillierte Verfahren vor Unvorhergesehenem zu schützen. Das Bedürfnis, Änderungen formal zu erfassen, führt daher in den deutschen Mustern teilweise zu fraglichen Detailregelungen, die bisweilen wie hilflose Versuche wirken, die Veränderungen innewohnende Unsicherheit zu vermeiden. Angemessene Reaktionsmechanismen für den Wandel der Anforderungen sind also weitgehend nicht vorgesehen.

C. Einbindung des Softwarebestellers - Mitwirkungspflichten

Die Leistungsbestimmung in Softwareentwicklungsverträgen ist ohne Mitwirkung des Softwarebestellers nicht möglich. Dies hat sich im ersten Kapitel aus informationstechnischer Sicht gezeigt und wurde im fünften Kapitel von zahlreichen Autoren aus juristischer Sicht bestätigt. Hier soll daher die Frage beantwortet werden, ob und ggf. wie der Softwarebesteller

⁵³⁷ In einem vom Verfasser im Rahmen der Untersuchung geführten Interview unterstrich ein Interviewpartner, dass die Softwarehäuser in den (19)80er Jahren infolge der Situation von Angebot und Nachfrage eine deutlich bessere Verhandlungsposition gehabt hätten.

⁵³⁸ Ausgegangen wurde dabei von der (manchmal illusorischen) Vorstellung, der Vertrag werde als eine gelebte Vereinbarung, die das Rückgrat des Projektes darstellt, und nicht als das mehr oder weniger überflüssige Produkt der Rechtsabteilungen bzw. juristischen Berater verstanden.

in beiden Rechtsordnungen und entsprechenden Verträgen eingebunden wird.

In den *restatements* wird eine Andeutung im Hinblick auf Mitwirkungspflichten im Kommentar dahingehend formuliert, dass eine *failure to cooperate in the other party's performance as bad faith* i.S. des § 205 ausgelegt werden kann⁵³⁹. Geht man von einer Anwendbarkeit des U.C.C. aus, so findet sich eine Norm, die ebenfalls von *cooperation* spricht, in § 2-311⁵⁴⁰. Es wird der Grundstein für eine Entschuldigung möglicher Verspätung bzw. aus Vertragsbruch herzuleitendem Schadensersatz gelegt⁵⁴¹.

Die Rechtsprechung sanktioniert demgemäß das Ausbleiben von Mitwirkungshandlungen im Einzelfall, wie oben⁵⁴² dargestellt, wobei dies auch vor dem Hintergrund des Grundsatzes geschieht, dass Rechtsfolgen aus einem Vertragsbruch nur geltend machen kann, wer selbst vertragstreu ist.

Das BGB fordert nunmehr in § 241 Abs. 2 ausdrücklich eine gegenseitige Rücksichtnahme⁵⁴³ und gibt in § 324 ein korrespondierendes Rücktrittsrecht,

⁵³⁹ Rs. 2d § 205: **Duty of Good Faith and Fair Dealing**

2. *Every contract imposes upon each party a duty of good faith and fair dealing in its performance and its enforcement.* S.o. Kapitel 3, S. 45f und 51ff.

⁵⁴⁰ **§ 2-311 U.C.C. Options and Cooperation Respecting Performance.**

2. (1) *An agreement for sale which is otherwise sufficiently definite (subsection (3) of Section 2-204) to be a contract is not made invalid by the fact that it leaves particulars of performance to be specified by one of the parties. Any such specification must be made in good faith and within limits set by commercial reasonableness.*

3. (2) *Unless otherwise agreed specifications relating to assortment of the goods are at the buyer's option and except as otherwise provided in subsections (1)(c) and (3) of Section 2-319 specifications or arrangements relating to shipment are at the seller's option.*

4. (3) *Where such specification would materially affect the other party's performance but is not seasonably made or where one party's cooperation is necessary to the agreed performance of the other but is not seasonably forthcoming, the other party in addition to all other remedies*

5. (a) *is excused for any resulting delay in his own performance; and*

6. (b) *may also either proceed to perform in any reasonable manner or after the time for a material part of his own performance treat the failure to specify or to cooperate as a breach by failure to deliver or accept the goods.*

⁵⁴¹ Dabei sollte nicht übersehen werden, dass es bei der Regelung des U.C.C. in erster Linie darum geht, die Verantwortlichkeit für die Konkretisierung eines Rechtsgeschäftes zuzuordnen. Demgemäß wird der angesprochene Fragenkreis auch unter dem Gesichtspunkt der *Indefinitness* diskutiert, vgl. Calamari/Perillo, a.a.O., 2.9, S. 50ff.

⁵⁴² Kapitel 3 D II.

⁵⁴³ Das Schuldverhältnis kann nach seinem Inhalt jeden Teil zur Rücksicht auf die Rechte, Rechtsgüter und Interessen des anderen Teils verpflichten.

wenn diese Pflicht verletzt wird und ein Festhalten am Vertrag unzumutbar ist.

Für den Werkvertrag und den nach § 651 S.3 BGB modifizierten Kaufvertrag ist eine Mitwirkungspflicht ausdrücklich bestimmt^{544, 545}. Auch hier hatten die Gerichte das Ausbleiben von Mitwirkungshandlungen in Einzelfällen zu beurteilen.

Das Fehlen qualifizierten Personals für das Projekt auf der Seite des Softwarebestellers (welches vor allem durch die Lieferung von Informationen zur Leistungsbestimmung beitragen kann) wurde von Gerichten in Deutschland und den USA als eine Verletzung der Mitwirkungspflicht angesehen⁵⁴⁶.

Seitens Gerichten beider Rechtsordnungen wird davon ausgegangen, das Softwarehaus habe die erforderlichen Mitwirkungshandlungen einzufordern, sei es durch das Verlangen nach Vorlage von Material oder der Teilnahme an Besprechungen. Wenn Spezifikationen laut Vertrag durch den Softwarebesteller zu erbringen sind, so verletzt er nach deutscher und US-amerikanischer Rechtsprechung seine Mitwirkungspflicht, wenn er es unterlässt sie vorzunehmen.

Eine bedeutende Einschränkung kann sich nach amerikanischer und deutscher Rechtsprechung aus Beratungspflichten des Softwarehauses ergeben. In einem von starkem Wissensgefälle geprägten Vertragsverhältnis wird Aufklärung und aktive Ermittlung von Informationen durch das Softwarehaus als selbstverständlich betrachtet. Es zeichnet sich allerdings in jüngerer Zeit die Tendenz ab, das Bestehen dieses Gefälles nicht mehr grundsätzlich anzunehmen, sondern vielmehr softwarespezifische Erfahrungen auch beim Softwarebesteller vorauszusetzen.

⁵⁴⁴ **§ 642 BGB Mitwirkung des Bestellers**

(1) Ist bei der Herstellung des Werkes eine Handlung des Bestellers erforderlich, so kann der Unternehmer, wenn der Besteller durch das Unterlassen der Handlung in Verzug der Annahme kommt, eine angemessene Entschädigung verlangen.

(2) Die Höhe der Entschädigung bestimmt sich einerseits nach der Dauer des Verzugs und der Höhe der vereinbarten Vergütung, andererseits nach demjenigen, was der Unternehmer infolge des Verzugs an Aufwendungen erspart oder durch anderweitige Verwendung seiner Arbeitskräfte erwirbt kann.

§ 643 BGB Kündigung bei unterlassener Mitwirkung

(1) Der Unternehmer ist im Falle des § 642 berechtigt, dem Besteller zur Nachholung der Handlung eine angemessene Frist mit der Erklärung zu bestimmen, dass er den Vertrag kündigt, wenn die Handlung nicht bis zum Ablauf der Frist vorgenommen werde.

(2) Der Vertrag gilt als aufgehoben, wenn nicht die Nachholung bis zum Ablauf der Frist erfolgt.

⁵⁴⁵ Siehe auch oben, Kapitel 5, V.

⁵⁴⁶ S .o. Kapitel 3, D II; Kapitel 5 II 5 b).

Gleichwohl legen die Gerichte, in Deutschland ausgeprägter als in den USA, große Zurückhaltung an den Tag, wenn es darum geht, die Notwendigkeit eines aktiven Parts des Softwarebestellers anzuerkennen.

Dies spiegelt sich auch in den Mustern wieder. Die an den Interessen des Softwarebestellers orientierten amerikanischen und deutschen Muster verlagern das Risiko einer - seitens des Softwarebestellers unerwünschten - Leistungsbestimmung weitestgehend auf das Softwarehaus.

Mitwirkungspflichten finden sich (nur) in den an den Interessen des Softwarehauses orientierten amerikanischen Mustern und in einem Praxisbeispiel. In den deutschen Mustern und Praxisbeispielen erschöpft sich die Pflicht des Softwarebestellers ganz überwiegend im Bereitstellen von (technischen) Ressourcen.

Während die deutschen Muster und Praxisbeispiele Personalfragen, also z.B. welche Mitarbeiterqualifikationen verlangt und v.a. wie sie aufrecht erhalten werden, ganz überwiegend keine Bedeutung beimessen, werden sie bei den softwarebestellerfreundlichen amerikanischen Mustern eingehend geregelt⁵⁴⁷.

Mitwirkungspflichten sind also in beiden Rechtsordnungen vorgesehen und in verschiedenen Gerichtsentscheidungen konkretisiert. Vom Softwarebesteller wird allerdings nur dann eine aktive Rolle verlangt, wenn dies ausdrücklich vertraglich vereinbart wurde und er über ausreichendes Know-How verfügt. Diese Tendenz ist im deutschen Recht ausgeprägter als im amerikanischen. Besonders auffällig stellt sie sich in an den Interessen der Softwarebesteller ausgerichteten Mustern bzw. Praxisbeispielen dar.

Auch hier zeigt sich ein quer durch die Rechtsordnungen verlaufender Bruch zwischen informationstechnisch für sinnvoll erachteten und vertraglich vorgesehenen Anforderungen an die Mitarbeit des Softwarebestellers. Der Softwarebesteller wird vielfach als Auftraggeber in der Rolle eines übergeordneten Geldgebers gesehen und seine Mitarbeit auf Lippenbekenntnisse bei fehlender konkreter Projekteinbindung reduziert.

⁵⁴⁷ Kapitel 4 A IV.

Hier zeigt sich ein weiteres nicht unerhebliches Gefährdungspotential für den Projekterfolg. Dieses Potential fällt aus deutscher Sicht generell und aus amerikanischer Sicht dann größer aus, wenn an den Interessen der Softwarebesteller orientierte Muster betrachtet werden.

D. Qualitätskriterien

Im amerikanischen Recht unterscheiden sich die rechtlichen Vorgaben für die Einstandspflicht des Softwarebestellers danach, ob der U.C.C. angewandt wird oder nicht⁵⁴⁸. Allgemein findet sich im Falle seiner Anwendung zunächst eine Regel, wonach die Waren (*goods*) dann zur Vertragserfüllung geeignet sind, wenn sie mit den vertraglichen Verpflichtungen (*obligations under the contract*) übereinstimmen, also die vereinbarte Beschaffenheit aufweisen⁵⁴⁹. Dies wird präzisiert, indem Zusagen und Bestätigung von Fakten, eine Beschreibung, Beispiele oder Modelle der Waren eine Einstandspflicht (*express warranty*), d.h. einen Erfüllungsmaßstab begründen⁵⁵⁰. Im Hinblick auf Softwareentwicklungsverträge hat zudem die Regel Bedeutung erlangt, wonach ein Käufer, der sich auf die Fähigkeiten und das Urteil des Verkäufers bei der Auswahl der Waren verlässt, soweit der Verkäufer von einem bestimmten Zweck, für den die Waren benötigt werden, wissen musste, erwarten kann, dass sie für diesen Zweck auch geeignet sind⁵⁵¹. Die Gerichte berücksichtigen in jedem Einzelfall, unabhängig von der Anwendbarkeit des U.C.C., die durch den Vertrag, dessen Durchführung und ggf. entsprechende Handelsbräuche begründeten Pflichten im Detail, was in

⁵⁴⁸ Dies gilt zunächst hinsichtlich der Frage, ob überhaupt erfüllt wurde oder nicht. Nach der *perfect tender rule* kann der Käufer die Ware zurückweisen, wenn sie auch nur geringfügig von der vertraglichen Vereinbarung abweicht (§ 2-601 U.C.C.). Diese Regel ist jedoch einiger Kritik ausgesetzt, und es sind zahlreiche Ausnahmen im Code selbst und durch Gerichtsentscheidungen vorgenommen worden, u.a. für den Fall, dass Waren (wie bei Softwareentwicklungsverträgen die Software) speziell für den Käufer gefertigt wurden, vgl. Calamari/Perillo, 11.20, S. 421. Demgemäß hat sich in dem untersuchten Case Law keine Partei jemals auf die *perfect tender rule* berufen, vgl. auch oben Kapitel 3, D. I. (S. 50). Bei Nichtanwendbarkeit des U.C.C. ist der Maßstab „due care“, s.o. Kapitel 3 C. I. 4. (S. 46).

⁵⁴⁹ § 2-106 (2) U.C.C.

⁵⁵⁰ § 2-313 (1) (a), (b) und (c) U.C.C.

⁵⁵¹ § 2-315 U.C.C., Implied Warranty: Fitness for Particular Purpose, eine Einstandspflicht, die allerdings abbedungen werden kann, wie in Verträgen teilweise vorgesehen, vgl. oben Kapitel 3, C. I.4. (S. 46).

der jüngeren Rechtsprechung zu günstigeren Entscheidungen für die Softwarehäuser geführt hat⁵⁵².

Woran die geschuldete Qualität zu messen ist, richtet sich im deutschen Recht in erster Linie nach dem schuldrechtlichen Vertragstyp, ergänzt durch den allgemeinen Grundsatz von Treu und Glauben⁵⁵³. Sowohl im Kauf- wie im Werkvertrag ist Maßstab die „vereinbarte Beschaffenheit“⁵⁵⁴. Ist die Beschaffenheit nicht vereinbart, so ist nach der „nach dem Vertrag vorausgesetzten“ Eignung zur Verwendung⁵⁵⁵ zu fragen und als letzte Alternative danach, ob das Werk bzw. die Sache sich für die „gewöhnliche Verwendung eignet und eine Beschaffenheit aufweist, die bei Werken (Sachen) der gleichen Art üblich ist und die der Besteller nach der Art des Werkes (der Sache) erwarten kann.“

Verschiedene Gerichte sowie einige Stimmen in der Literatur gehen davon aus, Fehlerfreiheit sei keine bei Individualsoftware zu erwartende Beschaffenheit⁵⁵⁶.

In beiden Rechtsordnungen ist die Bestimmung der Einstandspflicht von der Antwort auf eine Vorfrage, nämlich der Einordnung von Software in das Rechtssystem, abhängig. Ebenso ist die vertragliche Vereinbarung stets der Ausgangspunkt der Betrachtung. Der Wille der Parteien ist für den Qualitätsmaßstab ausschlaggebend. Bei der Bestimmung der geschuldeten Qualität ist in beiden Rechtsordnungen eine Tendenz festzustellen, keinen allzu strengen Maßstab anzulegen.

Die in den Mustern und Praxisbeispielen formulierten Maßstäbe sind zunächst in einer Übersicht dargestellt:

⁵⁵² S.o., Kapitel 3, D. II S 53.

⁵⁵³ Kodifiziert im BGB in § 242 Leistung nach Treu und Glauben: Der Schuldner ist verpflichtet, die Leistung so zu bewirken, wie Treu und Glauben mit Rücksicht auf die Verkehrssitte es erfordern.

⁵⁵⁴ § 434 Abs.1 S.1 BGB, § 633 Abs. 2 S.1 BGB.

⁵⁵⁵ § 434 Abs. 1 S.2 Nr. 1 BGB, § 633 Abs.2 S.2 Nr. 1 BGB.

⁵⁵⁶ S.o. Kapitel 5, B. II. 2. (S. 111f).

Maßstab der Leistungserbringung	
Deutsche Muster	
Vereinbarte bzw. vertraglich vorausgesetzte Beschaffenheit (so oder ähnlich in 6 Mustern)	
Aktueller Stand des Software Engineering und der Grundsätze ordnungsgemäßer Berufsausübung	
Lizenzprodukt erfüllt die Hauptfunktionen im Wesentlichen und entspricht den anerkannten Regeln der Technik	
Stand der Technik bei Vertragsabschluss; Verfahrensnormen nach DIN/ISO 9000ff. oder andere, in Hinblick auf die Qualitätssicherung gleichwertige Verfahrensregeln; Grundsätze ordnungsmäßiger Datenverarbeitung; Vorgaben der Bundesanstalt für Finanzdienstleistungsaufsicht, Erstellung so, dass die Software die Anforderungen an Zeitverhalten, Ergonomie, Fehlertoleranz, Wartbarkeit (auch durch Dritte) und Interoperabilität erfüllt und durch möglichst einfache Schnittstellen mit der angrenzenden Standardsoftware verbunden ist (Sicherung der Kompatibilität zu künftigen Ständen der Standardsoftware).	
Deutsche Praxisbeispiele	
Software kann nicht mit Sicherheit frei von Fehlern sein, <i>aber</i> bei EDV-gestützten Vorgängen an allen in Beilage 2 definierten Anlagen treten bei typischer Lastsituation zur Tageszeit der größten Last in 95 % aller Fälle Antwortzeiten unter drei Sekunden auf. Das System ist so gestaltet, dass es leicht erlernt und bedient werden kann; die Software wird gemäß den Programmierrichtlinien des Softwarehauses (Beilage 5) erstellt und gewartet (<i>es folgen zahlreiche Detailanforderungen funktioneller Art</i>)	
Die gesamte Software ist Jahr 2000 und EURO-fähig. Die vom System erzeugten Daten sind korrekt, soweit das Softwarehaus (AN) die Stammdaten ordnungsgemäß angelegt hat und keine Eingriffe in die vom AN gelieferten Systemkomponenten vorgenommen wurden. Der Nachweis für fehlerhafte Berechnungen muss vom Softwarebesteller (AG) geführt werden. Für vom AG vorgenommene Eingriffe übernimmt der AN keine Haftung.	
US-Muster (<i>die Parteibezeichnungen wurden angepasst</i>)	
Development Services and the Software will strictly comply with the descriptions (including performance capabilities, completeness, specifications, configurations, and function) that appear in the Scope of Work or in any Change Orders.	
Specifications will represent an acceptable implementation of the Functional Requirements. Softwarehaus warrants that for a period of twelve (12) months after XXXXX's first commercial shipment of the Product excluding shipments for beta testing, (the "Warranty Period"), the Product will perform in accordance with the Specifications. (...)	
The Subject Programs shall conform in all material respects to the Specifications and Performance Standards (except for subsequent modifications made at Softwarebesteller's request).	
The software will perform substantially in accordance with the specifications for the software	
The Management System will substantially conform to the specifications contained in Schedule „A“, as modified by the Modifications listed in Schedule „B“ (...)	
Application shall comply with Softwarebesteller's requirements as disclosed in the Detailed Design Specifications. (...)	

<i>Maßstab der Leistungserbringung</i>
Softwarehaus, as an expert in the field, warrants that: the system shall meet Softwarebesteller's needs (as defined in the approved Design Document and in Exhibit „A“) and shall be substantially free from programming errors and defects in workmanship and materials. (...)
<i>US-Praxisbeispiele</i>
The System shall substantially perform in accordance with all applicable Exhibits and shall be free of any defects which substantially affect the performance of the SMS System, as developed under the Prior Agreement. Softwarehaus represents and warrants that those MAS System enhancements developed by Softwarehaus in furtherance of this Agreement shall be free of any defects which substantially affect the performance of the SMS System, as updated by the SMS System Enhancements.
Softwarehaus shall provide such warranties for the repair of errors or defects in the Work as may be provided in the Authorization Letter applicable to such Work
Softwarehaus represents and warrants to Softwarebesteller that: (a) the System will be capable of fulfilling all of Softwarebesteller's current and reasonably anticipated needs in an efficient and commercially desirable and competitive manner; (b) there are no problems or defects, including viruses, known to Softwarehaus in the System; (c) the System does not contain any limitations which would render the System unsuitable for use by Softwarebesteller in processing its own applications and those of its current divisions, subsidiaries or affiliates, or presently contemplated future divisions, subsidiaries or affiliates; (d) the System will operate and conform in all respects during the term hereof in accordance with the Performance Standards; (e) the System will meet all applicable local, state and federal codes, laws, regulations and tariffs, and Softwarehaus has no knowledge of any pending or proposed change in such codes, laws, regulations or tariffs which the System as installed would not meet.

Zunächst fällt hinsichtlich der deutschen Muster auf, dass sie sich zum großen Teil mit der Verwendung des Wortlauts des Gesetzes begnügen. Soweit sie darüber hinausgehen, suchen sie technische Maßstäbe zu referenzieren. Von einer Ausnahme abgesehen, ist ein Zugeständnis an einen eingeschränkten Fehlerbegriff für Individualsoftware nicht vorhanden. In zwei Praxisbeispielen findet sich ein technischer Ansatz, wobei in einem Fall die Fehlerhaftigkeit von Individualsoftware ausdrücklich erwähnt wird.

Der Befund bezüglich der amerikanischen Muster ergibt deutliche Unterschiede. Die Bezugnahme auf eine gesetzliche Formulierung findet nicht statt. Ebenso wenig wird auf allgemeine technische Maßstäbe verwiesen. Als Anker dienen vielmehr, mit einer Ausnahme, stets die Spezifikationen. Einem eingeschränkten Fehlerbegriff wird in fünf Mustern

und einem Praxisbeispiel Rechnung getragen. In einem Muster findet sich andererseits die Pflicht des Softwarehauses, nicht nur fehlerfreie Software zu liefern, sondern damit auch die Bedürfnisse des Softwarebestellers zu erfüllen.

Im Ergebnis zeichnen sich die amerikanischen Muster und Beispiele weitgehend durch einen Pragmatismus aus, der den deutschen Mustern fremd zu sein scheint. Dies ist bezüglich des Fehlerbegriffes insbesondere deswegen verwunderlich, weil selbst die deutsche Rechtsprechung sich diesbezüglichen Zugeständnissen nicht versperrt. Auf den Stand der Technik wird möglicherweise verwiesen, um zu verhindern, dass ggf. die Gerichte einen Ausführungsstandard mittlerer Art und Güte⁵⁵⁷ zugrunde legen. Nähmen die Gerichte allerdings den Verweis insbesondere auf das Software Engineering (als die die Softwareentwicklung beschreibende informationstechnische Disziplin) ernst, kämen mit hoher Wahrscheinlichkeit Ergebnisse zustande, die sich die Unterzeichner der Verträge nicht träumen ließen.

Bezüglich der Ausgangsfrage ist daher festzustellen, dass in deutschen Mustern teilweise keine angemessenen Qualitätskriterien für die Beurteilung der Leistung gewählt und die mangelnde Eignung der Fehlerfreiheit ganz überwiegend nicht berücksichtigt wird. Die amerikanischen Muster bilden dazu in der Mehrzahl einen erfreulichen Kontrast.

E. Vergütung

Der Befund aus Mustern und Praxisbeispielen offenbart vergleichbare Konzepte: In den amerikanischen finden sich, mit Ausnahme eines Praxisbeispiels, ausschließlich zeitunabhängige Vergütungsregeln (Fixpreise). Auch in den deutschen sind, soweit nicht offen gelassen, ausschließlich Festpreisvereinbarungen anzutreffen.

⁵⁵⁷ Wie in § 243 Abs.1 BGB für Gattungsschulden vorgesehen.

Kapitel 8: Schlussfolgerungen

A. Strukturelle Unterschiede – Spielräume und ihre Verwendung

Die Leistungsbestimmung in Softwareentwicklungsverträgen vollzieht sich in einem von beiden Rechtsordnungen weitgehend unbeschränkten Rahmen.

Diese Freiheit erstreckt sich allerdings - vor allem im deutschen Recht - nicht auf das Unterlassen der Leistungsbestimmung. Wenn die Vertragsparteien einen Gesichtspunkt nicht geregelt haben, z.B. ob eine Dokumentation zu erstellen ist oder ein Pflichtenheft zu liefern, so wird eine entsprechende Pflicht konstruiert. Die Erweiterung der Leistungsbestimmung wird dann durch die Gerichte vorgenommen.

Haben die Parteien dagegen eine Vereinbarung getroffen, z.B. das Pflichtenheft werde vom Softwarebesteller vorgelegt, so schließt dies eine Intervention der Gerichte regelmäßig aus.

Die Spielräume können sehr unterschiedlich genutzt werden, wie im folgenden zu zeigen ist. Dabei ist zunächst hinsichtlich der für das deutsche Recht grundlegenden Weichenstellung bzgl. der Einordnung von Software Stellung zu beziehen.

I. Software ist keine Sache

Software ist keine Sache. Dieser so einfache Satz ist von Teilen der deutschen Literatur so schwer nachzuvollziehen. Dabei ist schon die Art und Weise, wie Begründungen für die Sacheigenschaft gesucht werden, ein deutlicher Hinweis auf die Fragwürdigkeit dieser Ansicht. Warum müssen abenteuerliche Konstruktionen⁵⁵⁸ gewählt werden, wo es viel einfacher ist, die Grenzen des Sachbegriffes zu überschreiten?

⁵⁵⁸ Wie etwa, Software diene der "Konditionierung der Hardware", Braun, in: BB 1992, S. 154 (157).

Die Genesis des Begriffes Software beruht auf dem Gegensatzpaar Hardware und Software. Hardware ist dabei das, was man anfassen kann, Software das, was sich der sinnlichen Wahrnehmung entzieht, also eben keine Sache⁵⁵⁹.

Zweifellos erfüllen die Träger der Software die Sacheigenschaft. Dies gilt allerdings auch für Träger von Film, Musik und Duft, weshalb Film, Musik und Duft selbst keineswegs als Sachen anzusehen sind. Software entfernt sich bei diesem Vergleich noch weiter vom Sachbegriff, da Film, Musik und Duft direkt zur sinnlichen Wahrnehmung durch Auge, Ohr und Nase bestimmt und geeignet ist. Software selbst ist mit den Sinnen dagegen nicht wahrnehmbar und auch nicht dafür gedacht. Was sinnlich wahrnehmbar sein kann, sind allenfalls die Ergebnisse der Software.

Im deutschen wie im amerikanischen Recht ist die Argumentation anzutreffen, Medium und Programm seien unentwirrbar verbunden⁵⁶⁰, und weil dies so sei, müsse die Sacheigenschaft bejaht werden. Wie die Gegner der Sacheigenschaft dargelegt haben, ist diese Behauptung angesichts neuer Technologien und darauf basierender Konzepte wie Netzwerke (lokale und globale wie das Internet) und z.B. Application Service Providing unzutreffend. Doch selbst wenn eine Untrennbarkeit angenommen wird, so ist die daraus gezogene Schlussfolgerung nicht zwangsläufig, ja nicht einmal logisch. Gedanken (Programme) und Gehirn (Körper) bedingen einander, da der Mensch ohne Gehirn nicht denken kann. Sind Gedanken körperliche Gegenstände, weil das Gehirn ein körperlicher Gegenstand ist? Strom und Computer sind in gleicher Weise „untrennbar“ miteinander verknüpft. Der Computer ist ohne Energie ein Stück Metall, der Strom ohne Computer nicht wahrnehmbar. Kann Strom deswegen als körperlicher Gegenstand bezeichnet werden?

⁵⁵⁹ An dieses Kriterium wird für die Definition von Software angeknüpft. Vgl. Brockhaus Computer- und Informationstechnologie unter *Software*: „jede immaterielle (eben „weiche“) Komponente eines Computersystems, im Unterschied zur „greifbaren“ Hardware.“; Grieser/Irlbeck unter *Software*: „alles was im Gegensatz zur Hardware nicht physikalisch vorhanden, also nicht anfassbar ist“; Feretti unter *Software*: „Überbegriff für alle immateriellen Bestandteile eines Computers“.

⁵⁶⁰ S.o. Kapitel 3, C. I. 2. (S. 42) und Kapitel 5, B. I. 2. b) (S. 109 und Fn. 379).

Der Sourcecode eines Programms in seiner einfachsten Form sieht z.B. so aus:

```
PROGRAM Hallo Welt  
BEGIN  
Write Line „Hallo World“;  
END
```

Die Ausführung dieses Programms wird zur Bildschirmausgabe „Hallo World“ führen. Worin ist hier eine Sacheigenschaft zu sehen?⁵⁶¹

Der Gegenmeinung ist zuzugestehen, dass sie einen gewichtigen Fürsprecher auf ihrer Seite hat. Dies könnte in der rechtlichen Diskussion einen Ansporn zur Kontroverse bilden, doch handelt es sich bei dem Fürsprecher um das höchste deutsche Gericht. Der von diesem eingeschlagene Weg, auch wenn er nicht auf sorgfältiger begründeter Entscheidung beruht⁵⁶², wird vielfach von der Zivilgerichtsbarkeit nachvollzogen. Allein dies verleiht ihm einiges an faktischem Gewicht. Es ändert jedoch nichts an der Fehleinschätzung. Der gerichtlichen Praxis kann zugute gehalten werden, dass sie eine Entscheidung treffen muss und dabei verständlicherweise auf vertraute Konzepte zurückgreifen möchte. Dieser pragmatische Ansatz kommt auch sehr gut in einem amerikanischen Urteil zum Ausdruck. Vertragsschließende könnten danach berechtigterweise davon ausgehen, zumindest *ein Gesetz* sei anwendbar⁵⁶³. Es war dabei allerdings mit einer Entscheidung anderer Tragweite, als die deutschen Gerichte sie zu gewähren hätten, konfrontiert. Die Anwendbarkeit des U.C.C. insgesamt hing von der Einordnung der Software als *good* ab. Bei einer Entscheidung gegen die Sacheigenschaft hätte das Gericht also ohne Gesetz „dagestanden“. Einem deutschen Gericht wäre dann immer noch die Möglichkeit gegeben, auf Vertragstypen auszuweichen, die eine

⁵⁶¹ Der mögliche Einwand, es handle sich um ein Programm und nicht um Software, soweit man dieser Unterscheidung überhaupt folgt, greift nicht. Wenn das Programm in eine computerlesbare Sprache übersetzt und anschließend zum Herunterladen auf einem Webserver bereitgestellt wird, handelt es sich dann nicht um Software? Der Erwerb einer Vielzahl von Programmen, einschließlich umfangreicher Betriebssysteme, findet genau auf diese Weise statt. Wenn hier Programme veräußert werden und nicht Software, worin besteht dann der Unterschied?

⁵⁶² Nachweise s.o. in Kapitel 5 B. I. 2. (S. 107).

⁵⁶³ S.o. Kapitel 3 C. I. 2. (S. 42).

Sacheigenschaft nicht voraussetzen, wie z.B. den Werk- oder Dienstvertrag, oder vorhandene Regeln analog anzuwenden.

II. Lösungen im Begriffssystem und Erfahrungssystem

Die Frage „Wenn Software keine Sache ist, was ist sie dann?“ kann sehr einfach beantwortet werden. Software ist Software! Es ist nicht notwendig, auch wenn der „*civil lawyer*“ dies als schwer erträglich empfindet, Software der vorgegebenen Begriffswelt der Gesetze zuzuordnen. Der Rechtsvergleich kann an dieser Stelle helfen, sich von den Begriffsvorgaben zu lösen.

Das deutsche Recht schöpft – strukturell - aus dem Gesetz als primäre Rechtsquelle. Wie sich beispielsweise bei der untergerichtlichen Rechtsprechung zur vertraglichen Einordnung der Softwareentwicklung zeigt, reicht allerdings eine Stellungnahme des BGH häufig dazu aus, dem dort vorgegebenen Weg ohne weitere Begründung zu folgen. Gleichwohl macht sich der Strukturunterschied zum amerikanischen Recht beispielsweise bei Gesetzesänderungen wie der Schuldrechtsreform bemerkbar. Er wird für Softwareentwicklungsverträge auch sehr deutlich bei der Frage nach der Einordnung in das besondere Schuldrecht spürbar. Die rechtliche Entscheidung scheint hier sehr in einem System von Begriffen, wie denen der Sache oder des Werkvertrages, gefangen. Zwar stellt sich auch im amerikanischen Recht z.B. die Frage nach der Sacheigenschaft von Software, doch wird sie weitaus weniger problematisiert und spielt in der Rechtsprechung eine untergeordnete Rolle. Pointiert formuliert könnte man insbesondere aus der anhaltenden Diskussion in der Fachliteratur den Eindruck gewinnen, der deutsche Jurist verbringe beträchtliche Zeit damit, die „richtige Schublade“ zu suchen, er sei mit der Etikettierung der Bäume so beschäftigt, dass er den Blick für das Ganze, den Wald, verliere.

Dieser Schein trügt. Es mag im Einzelfall zutreffen, dass der Aufwand für die Suche nach dem „richtigen Gewand“ unverhältnismäßig hoch ausfällt. Er ist jedoch überwiegend von dem Wunsch nach einem angemessenen Ergebnis geprägt. Der Blick über den Zaun, im Fall der vorliegenden Untersuchung

über den Atlantik, führt jedoch vor Augen, wie begrenzt eine in der Dogmatik verhaftete Betrachtungsweise bleibt. Sie dreht sich im Kreise der selbstgeschaffenen Begriffe und Systeme, und es fällt ihr daher besonders schwer, Veränderungen zur Kenntnis und in ihre Denkweise aufzunehmen.

Es ist deutlich einfacher und hilfreicher, den Fokus von der Frage „Was ist Software?“ auf die Frage „Wie können die im Zusammenhang mit Software auftretenden rechtlichen Probleme gelöst werden?“ zu lenken. Für diese Untersuchung kann die Frage weiter präzisiert werden: Welche rechtlichen Regeln unterstützen Softwareentwicklungsverträge?

Diese Formulierung ist bewusst gewählt. Sie beruht auf dem Standpunkt, dass das Recht dem Menschen dienen muss, und nicht der Mensch dem Recht. Die Beschäftigung mit der rechtlichen (Auffang-) Lösung oder besser Begleitung eines tatsächlichen Phänomens sollte eine Frage nach der optimalen Gestaltungsform für eine möglichst große Anzahl von Menschen sein. Eine Rechtsordnung, in der wir gewohnt sind, neue „Fälle“ in ein System geschriebener Gesetze einzuordnen, tut sich scheinbar mit Antworten schwerer als ein System, das eine Antwort „von Fall zu Fall“ geben kann. Doch auch dieser Schein trügt.

Ein am Gesetz ausgerichtetes Rechtssystem wie das deutsche lehrt eine möglichst präzise Subsumtion des Sachverhalts unter den vom Gesetzgeber vorgegebenen Tatbestand. Dabei trifft der Gesetzgeber auf die für alle Rechtsordnungen geltende Realität der wunderbaren Vielfältigkeit dieser Welt! Wie kann diese in das spröde Gewand von Rechtsnormen gekleidet werden? Für die Lösung dieser, eigentlich unlösbaren, Aufgabe bedient das Recht sich des „Tricks“ der Abstraktion.

Auf diese Weise werden Kinder, Greise, Frauen, Männer, gleich welcher Hautfarbe, Herkunft und sonstiger Daseinsumstände, zu „natürlichen Personen“. Blumen, Bäume, Häuser, Autos und Computer werden zu „Sachen“. Sprache, Geste, mitunter auch Schweigen wird zur „Willenserklärung“, Geben und Nehmen zum „Rechtsgeschäft“. Die Arbeit eines Schreiners, Automechanikers, einer Floristin oder Malerin, eines Computertechnikers oder einer Programmiererin wird zum „Werkvertrag“.

Die Abstraktion findet dabei bekanntermaßen im deutschen Recht dergestalt in Stufen statt, dass einem spezielleren Teil ein allgemeiner vorangestellt wird. Daraus ergibt sich für die Frage nach den geeigneten Rechtsregeln die Möglichkeit, eine Lösung auf der allgemeineren Ebene, dem allgemeinen Schuldrecht⁵⁶⁴ oder der Ebene des besonderen Schuldrechts zu suchen. Es ist weder erforderlich noch sinnvoll, als einzige Möglichkeit eine Zuordnung zu einem - vorhandenen - Typ des besonderen Schuldrechts zu sehen⁵⁶⁵.

Ein vergleichender Blick zum amerikanischen Recht erweist sich auch hier als hilfreich. Es gelingt den US-Gerichten durchaus, auch bei Nichtanwendbarkeit des U.C.C. zu einer Lösung zu gelangen. Selbst dort, wo er angewandt wird, finden die Gerichte keine dem Werkvertrag vergleichbare Regelung vor, was sie wiederum nicht darin hindert, sachgerechte Entscheidungen zu treffen.

Es wäre dabei sicher voreilig, den Grund hierfür in der vom Fallrecht geschulten Vorgehensweise der amerikanischen Richter zu suchen. Ebenso voreilig wäre es, das Prinzip der Abstraktion vorrangig dem deutschen Recht zuzusprechen. Die oben bereits mehrfach angeführten *Restatements of Law* sind ein Paradebeispiel für Abstraktion und stehen diesbezüglich den allgemeinen Teilen des deutschen BGB in nichts nach.

Auf den Punkt gebracht könnten die deutschen Gerichte also einen Softwareentwicklungsvertrag als Vertrag sui generis handhaben und eigene Rechtsregeln hierzu entwickeln. Ein solcher Schritt erfordert allerdings den Mut dazu, einer beengenden Dogmatik den Rücken zu kehren und der eigenen Überzeugung zu folgen. Letztendlich können auf diese Weise dann allerdings auch Signale an den Gesetzgeber gesendet werden.

Diese gesetzgeberische Initiative kann natürlich auch ohne gerichtlichen Anstoß ergriffen werden. In den USA ist dies bereits geschehen und mündete in den oben beschriebenen Entwurf des Uniform Computer Information Transaction Act (UCITA). Er erlangte u.a. aus Kompetenzgründen keine Geltung⁵⁶⁶. Gäbe es einen amerikanischen

⁵⁶⁴ Ein Weg, der z.B. von Hilty auf Basis einer Lizenzvertragslösung vorgeschlagen wird, in: MMR 2003, S. 3 (15 und Fn. 114).

⁵⁶⁵ Wovon jedoch die Mehrzahl der Fachbeiträge auszugehen scheinen. Der ganz überwiegende Teil der Diskussion verläuft entlang der Zuordnungsfrage: Werkvertrag oder Kaufvertrag, s.o., Kapitel 5 B. I. 2. (S. 106).

⁵⁶⁶ s.o. Kapitel 3 A. (S. 38 Fn. 141) und C. III. (S. 49).

Gesetzgeber, der zivilrechtliche Normen für die gesamte USA schaffen und verbindlich in Kraft setzen könnte, so wäre der UCITA höchstwahrscheinlich seit mehreren Jahren geltendes Recht.

An dieser Stelle können nunmehr die Vorteile des deutschen „Civil Law-Systems“ fruchtbar gemacht werden. Der deutsche Bundestag wäre ohne weiteres in der Lage, ein Computer- und Informationsgesetz zu beschließen⁵⁶⁷, einen neuen Vertragstypus zu schaffen, oder - als bescheidenste Lösung - Änderungen am bestehenden Gesetz, beispielsweise in Form einer expliziten Zuweisung der Softwareverträge zum Werkvertragsrecht vorzunehmen. An eine derart geänderte Rechtslage wären alle deutschen Gerichte gebunden, und es bestünde auch für die Kautelarjurisprudenz eine klare Vorgabe, die es zu beachten gilt und die als Wegweiser dienen kann.

Eine Lösung im Begriffssystem des deutschen Rechts ist einerseits schwieriger, weil sie sich im beständigen Ringen von notwendiger Abstraktion und gewünschter, bisweilen ersehnter Konkretisierung ihre Berechtigung für einen dauerhaften Platz erkämpfen muss⁵⁶⁸. Sie ist andererseits einfacher, weil sie nur in begrenztem Maße einer Vielzahl von u.U. divergierender Staatsinteressen ausgesetzt ist, die zudem in einem Staatsorgan, dem Bundesrat, gebündelt zu Tage treten. Der Rechtsvergleich offenbart also Chancen, die von einem deutschen Gesetzgeber ergriffen werden können.

III. Grenzen der Begriffssysteme

„The legal order is seen as something static. Law is viewed not as a process for the perception and resolution of problems, but as a set of established rules and institutions.“⁵⁶⁹ Dieses Urteil eines Common Law Juristen über das Civil Law ist nach den Erkenntnissen aus den untersuchten (Muster-) Verträgen in dieser Pauschalität sicher unzutreffend. In der

⁵⁶⁷ Er dürfte bei dem geringen politischen Potential der Regelung auch nicht mit größerem Widerstand des Bundesrates zu rechnen haben, der sich dann auch üblicherweise auf Anpassungen im Detail beschränkt.

⁵⁶⁸ Diesen „Kampf“ haben beispielsweise der Teilzeit-Wohnrechtsvertrag (§§ 481ff BGB), der Reisevertrag (§ 651a ff), der Darlehensvermittlungsvertrag (§§ 655a ff), und die Geschäftsbesorgungsverträge aus der Finanzwirtschaft (§§ 676a ff) gewonnen. Ihren „Sieg“ verdanken alle genannten Beispiele, bis auf den Darlehensvermittlungsvertrag, allerdings dem europäischen und nicht dem deutschen Gesetzgeber.

⁵⁶⁹ Merryman, *The Civil Law tradition*, a.a.O., S. 69.

Vertragsgestaltung kommt zum Ausdruck, wie das Recht in der (rechtlichen) Praxis verstanden und angewandt wird. Und hier zeigen sich nicht die Unterschiede, die nach dem vorstehenden Zitat zu erwarten wären. Deutsche und amerikanische Vertragsjuristen versuchen gleichermaßen, Lösungen für die komplexe Aufgabe der Steuerung von Softwareentwicklungsprojekten zu finden. Dabei mag der Versuch einer scharfen Trennung von Technik und Recht in manchen deutschen Mustern der rechtssystematischen Idee entspringen, zwei unabhängig voneinander bestehende Begriffswelten, sozusagen einen allgemeinen und einen speziellen Teil, zu schaffen. Der juristische Teil kann dann, wie in zumindest einem Beispiel anzutreffen, zur Abwicklung eines umfangreicheren Projektes schlicht ungeeignet sein. Doch gibt dies allenfalls einen Hinweis auf eine bisweilen vorzufindende Neigung, die im amerikanischen Rechtsraum seltener anzutreffen ist, nicht jedoch einen allgemeinen Befund. Überwiegend werden die Grenzen von Begriffssystemen erkannt.

Bedenklicher stimmt angesichts der Untersuchungsergebnisse die Einschätzung, dem „Civil Lawyer“ sei das Prinzip der Rechtssicherheit so wichtig, dass er bereit sei, die Einzelfallgerechtigkeit dafür zu opfern⁵⁷⁰. In dem Wunsch nach dieser Rechtssicherheit könnte eine Triebfeder für die oben beschriebene extensive rechtliche Ausformung von Änderungsszenarien in den deutschen Mustern erblickt werden. Hier gilt es allerdings, Rechtssicherheit in Verträgen und in Gesetzen zu unterscheiden. Die letztere hat nicht nur die Aufgabe, dem Normadressaten ein regelgerechtes Verhalten zu ermöglichen, sondern muss auch unter dem Gesichtspunkt der Gleichbehandlung bestehen. Gesetze gelten für alle, Verträge begründen dagegen nur Pflichten zwischen den Parteien. Dabei ist es die jeweilige besondere Situation der Parteien, die im Vertrag zu berücksichtigen ist. Einzelfallgerechtigkeit wird durch Anpassung des Vertrages an die besonderen Umstände gewährleistet⁵⁷¹. Soweit also, insbesondere nach den Erkenntnissen aus dem zweiten Kapitel, Änderungen ein bedeutender Bestandteil von Softwareentwicklungsverträgen sind, sollten für den Fall ihres Auftretens

⁵⁷⁰ Merryman, *The Civil Law Tradition*, a.a.O., S. 71:“(…) *the legal order frequently sacrifices the justice of the individual case to the demands of certainty in the law*“.

⁵⁷¹ Weshalb der deutsche Gesetzgeber in diesem Fall beispielsweise grundsätzlich keine Kontrolle nach den §§ 305c ff BGB vorsieht (Vorrang der Individualabrede - § 305cBGB).

auch umfassende Regelungen vorgesehen werden. Dies ist kein Tribut an das Verdikt der Rechtssicherheit, sondern eine Reaktion der Vertragspraxis auf die faktischen Gegebenheiten, mithin eine erfahrungs- und keine begriffsorientierte Vorgehensweise.

Es bleibt das Bemühen um einen allgemeinen Maßstab für Anforderungen an die Technik als Ausweis einer im deutschen Rechtssystem verhafteten Denkweise. Diese nur in den deutschen Mustern zu findende Eigenheit mag einem Denken in Begriffssystemen geschuldet sein. Die Absicht, informationstechnischen Gegebenheiten Rechnung zu tragen, ist erkennbar, bleibt jedoch im Detail zu ungenau und vermag daher insgesamt nicht zu überzeugen. Der Verweis auf den „Stand der Technik“ soll die magische Wirkung einer Qualitätssicherung entfalten. Dies kann ihm nicht gelingen. Er ist Ausdruck des Wunsches, zumindest einen Bezugspunkt zu der vertrauten juristischen Welt zu setzen, die den Anschein einer Subsumtionsmöglichkeit eröffnet. Das Projekt „Softwareentwicklung“ wird auf diese Weise nicht gefördert. Vielmehr kann sich bestätigt sehen, wer ein Spezifikum des Civil Law darin sieht, dass generelle Aussagen auf Kosten der Besonderheit des Einzelfalles überbetont werden⁵⁷².

IV. Erfolgsorientierte Fehlerbestimmung

Die Untersuchung hat gezeigt, dass Fehlerfreiheit als Erfüllungsmaßstab bei komplexer Software aus Sicht des Softwareengineering ungeeignet ist. Dies wird von Teilen der deutschen Rechtsprechung ebenso beurteilt und findet sich in der ganz überwiegenden Anzahl der amerikanischen Muster und Praxisbeispiele in Anlehnung an den in Literatur und Rechtsprechung weitgehend anerkannten Grundsatz, dass „*substantial performance*“ für eine vertragsgemäße Erfüllung ausreicht, wieder.

Dessen ungeachtet halten die deutschen Muster mit großer Mehrheit an dem Kriterium der Fehlerfreiheit fest.

Der Grund hierfür ist vermutlich in der Befürchtung zu finden, das Softwarehaus könne sich auf einfache Weise aus der Verantwortung für

⁵⁷² Merryman, *The Civil Law Tradition*, a.a.O., S. 72, der anmerkt, die traditionelle Sichtweise der civil-law-Rechtswissenschaftler verenge den Blick auf die (rechts-) dogmatische Frage nach dem, was „*ist*“, und vermeide die Frage danach, was „*sein soll*“.

ordentliche Qualität schleichen, wenn einmal ein Einfallstor für eine Begrenzung der Einstandspflicht geöffnet würde. Dem ließe sich aus Sicht des Softwarehauses, in diesem Fall aus Angst vor böswilligen Softwarebestellern motiviert, entgegenhalten, der Anspruch auf ein fehlerloses Produkt ließe jedes Softwareprojekt zu einem Va-banque-Spiel werden. Das Softwarehaus sei den Launen des Bestellers ausgeliefert, da dieser sich aus dem Pool der unvermeidbaren Fehler nach eigenem Gutdünken und zum eigenen Vorteil bedienen könne.

Auf Misstrauen gründende Vertragsformulierungen bilden keine solide Basis für ein Langzeitprojekt, wie z.B. einen Softwareentwicklungsvertrag. Setzt sich eine Vertragspartei über das Unwohlsein der anderen Vertragspartei in diesem Punkt hinweg, so ist der erste Schritt für einen späteren Konflikt bereits gegangen. In einem Streitfall wird es sehr wahrscheinlich nie darauf ankommen, ob eine theoretisch verlangte oder nicht erforderliche Fehlerfreiheit zu praktischen Konsequenzen führt⁵⁷³. Wie oben dargestellt, ist es nach der Judikatur bereits keineswegs selbstverständlich, dass überhaupt ein präzises Leistungsergebnis mit hinreichender Sicherheit ermittelt werden kann. Haben die Parteien eine klare Vereinbarung getroffen, so bildet diese den Maßstab für eine ggf. später notwendig werdende Kontrolle. Der Fehler ist dann, wie auch vom Gesetzgeber vorgesehen, durch die vertragliche Vereinbarung definiert. Dies kann eine Einstandspflicht über die anerkannten Regeln der Technik hinaus zu Folge haben⁵⁷⁴. Liegt eine präzise Leistungsbestimmung vor, ist diese das Maß aller Dinge. Sie fixiert die „*four corners of the instrument*“, zwischen denen die geschuldete Leistung Gestalt annimmt. Auf unausgesprochene Erwartungen, vorausgesetzte Bedingungen oder Einschränkungen kommt es nicht an. Unjuristisch bedeutet dies nichts anderes als die triviale Feststellung: „Halte Dein Versprechen“.

Der Gewinn aus der rechtsvergleichenden Betrachtung sollte für das deutsche Recht aus der prinzipiell großzügigeren Sichtweise des

⁵⁷³ So zutreffend auch Redeker, der betont, dieses Problem habe sich in der Praxis nie gestellt, CR 2005, a.a.O., S. 700 (703).

⁵⁷⁴ BGH, U.v. 15.10.02, in: NJW 2003, S. 200 (201): „*Der Auftraggeber hat die Entstehung eines mangelfreien, zweckgerechten Werkes zu gewährleisten. Entspricht die Leistung nicht diesen Anforderungen, so ist sie fehlerhaft, und zwar unabhängig davon, ob die anerkannten Regeln der Technik eingehalten worden sind. Ausschlaggebend ist allein, dass der Leistungsmangel zwangsläufig den angestrebten Erfolg beeinträchtigt.*“ Zuletzt auch OLG Bremen, U.v. 07.07.04 - 4 U 64/03-, in: BauR 2005, S. 1679f. Das OLG Nürnberg, U.v. 23.06.05 -13 U 1943/02 – in: BauR 2005, S. 1680 formuliert dies so: „*Ergibt die Auslegung ein bestimmtes Vertragsoll, so ist das Werk mangelhaft, wenn die Sollbeschaffenheit nicht erreicht ist.*“

amerikanischen Rechts gezogen werden. *Substantial performance* in diesem Sinne heißt, die zentralen Funktionen, die die Software erfüllen soll, nicht über Detailfragen aus den Augen zu verlieren. Nach der Paretoregel⁵⁷⁵ verursachen die letzten 20 %, also die Spezialfälle, etwa 80 % des Aufwandes. Vielleicht wird deswegen davon ausgegangen, das Projekt sei etwa zur Hälfte fertig, wenn die Entwickler davon sprechen, es wäre bei einem Erfüllungsgrad von 90% angelangt⁵⁷⁶. Wenn dem Softwarehaus ein perfektionistischer und vermeintlich präziser Fehlerbegriff im Nacken sitzt, ist dem Projekt nicht gedient, und die Abnahmeprozedur wird leicht zum „Spießrutenlaufen“ für beide Vertragsparteien. Großzügigkeit ist nicht mit Nachgiebigkeit zu verwechseln, sondern stellt eine Konzentration auf das Wesentliche dar. Durch Kombination dieser wesentlichen Leistungsmerkmale mit einer präzisen Leistungsbestimmung kann eine tragfähige Basis gerade für hochkomplexe Projekte geschaffen werden.

⁵⁷⁵ s.o. Kapitel 3 A. III. 1. b) (S. 22).

⁵⁷⁶ Vgl. hierzu oben Kapitel 2 A. III. 3. b) (S. 31 Fn. 115) und die instruktive Darstellung bei De Marco, *The Deadline*, der zwischen Zeitplan und Ziel unterscheidet und empfiehlt, sie unterschiedlich zu definieren, S. 289f und S. 302.

Kapitel 9: Zusammenfassende Würdigung

A. Gesamtbefund

Die vorangegangene Untersuchung kann zu drei zentralen Themenkreisen verdichtet werden. Der erste betrifft das Verhältnis von Technik und Recht (dazu I.), der zweite die Beziehung zwischen Planung und Wirklichkeit (dazu II.), der dritte die Eignung von interessenorientierter (anstelle von projektorientierter) Vertragsgestaltung zur Optimierung des Projektverlaufs (dazu III.).

I. Technik und Recht – Social Engineering und Software Engineering

Die im Rahmen von Softwareentwicklungsverträgen auftauchenden Fragestellungen werden von dem Vertragsgegenstand und nicht von der Rechtsordnung, innerhalb derer sie beantwortet werden, geprägt. Die diagnostizierten Brüche verlaufen also nicht zwischen dem deutschen und dem amerikanischen Recht, sondern zwischen Technik und Recht. Die *praesumptio similitudinis*, also die Vermutung der Ähnlichkeit der Lösungsansätze bestimmter Sachverhalte bei der Rechtsvergleichung, findet sich an dieser Stelle bestätigt⁵⁷⁷. Dies gilt sowohl für die Bereiche, in denen eine rechtliche Erfassung der Realität im großen und ganzen gelingt, als auch dort, wo sie weitgehend versagt.

Das *social engineering*⁵⁷⁸ gelingt grundsätzlich bei der Konzeption von Verfahren zur Bestimmung eines präzisen Zieles. Die systematische Vorgehensweise der Juristen lässt sich hier offenbar gut mit der Vorgehensweise der Informatiker in Übereinstimmung bringen. Dabei sind allerdings zwei Einschränkungen zu beachten. Die erste betrifft ein juristisches Defizit. Die Geschwindigkeit, mit der sich die Technik und die damit verbundenen Verfahren ändern, kann durch die Rechtsordnung nicht nachvollzogen werden. Das v.a. in Form von Vertragsgestaltung zu Tage tretende Recht ist bei dem Versuch der Abbildung der ersten Phasenmodelle

⁵⁷⁷ Zweigert/Kötz, Einführung in die Rechtsvergleichung auf dem Gebiete des Privatrechts, 3. Aufl., Tübingen, 1996, S. 39.

⁵⁷⁸ Zweigert/Kötz, a.a.O., S. 45.

stecken geblieben. Zyklische Modelle, wie z.B. das Spiralmodell, scheinen das Recht zu überfordern.

Die zweite Einschränkung betrifft ein Defizit der (Informations-) Technik in der Praxis. Wie diese Praxis lehrt, wird allzu häufig gegen die erkannte und von Juristenseite unermüdlich angemahnte Regel, dass der Softwareentwicklungsvertrag auf eine präzise Leistungsbestimmung gegründet sein muss, verstoßen.

Das social engineering versagt in vielen Fällen bei Änderungen des Leistungsprogrammes (dazu sogleich unter II.) und bei der Gestaltung von Mitwirkungspflichten.

Die Gründe hierfür entspringen m.E. dem soeben angedeuteten Auseinanderfallen von technischer und rechtlicher Entwicklung. Bereits in einem Softwareprojekt mittlerer Größe⁵⁷⁹ arbeiten möglicherweise bis zu 300 Entwickler in bis zu 50 Teilprojekten über viele Monate zusammen, wobei räumliche Entfernungen keine Rolle spielen. Zur Veranschaulichung: Eine Arbeitsgruppe des (deutschen) Softwarehauses in Indien bereitet ein Modul vor, welches in den USA mit anderen Modulen zusammengefasst und in Deutschland durch ein drittes Team beim Softwarebesteller implementiert werden soll, wobei der Softwarebesteller in zwei dieser Arbeitsgruppen mit mehreren Mitarbeitern vertreten ist und durch ein drittes Team, welches in der Planungsphase maßgeblich beteiligt war, die Test- und Installationsphase vorbereitet und überwacht. Aus Sicht des BGB wird hier ein unbekannter „Vertragsgegenstand“ (Software) in nicht vorgesehener Form der Zusammenarbeit (aus Mitarbeitern von Auftraggeber und -nehmer zusammengesetzte Projektteams, möglicherweise in Kombination mit Subunternehmerbeteiligung) in unbekannter Vorgehensweise (zyklische Abwicklung) über einen langen Zeitraum (mindestens mehrere Monate, häufig mehrere Jahre) abgewickelt. Weyers hat als Ergebnis seiner eingehenden Untersuchung des Werkvertragsrechts herausgestellt, dass die gleichen Fragen in verschiedenen Kombinationen und mit geringen Abwandlungen bei den unterschiedlichsten Einzelverträgen immer wieder auftauchen⁵⁸⁰ und auf

⁵⁷⁹ S.o. Kapitel 2.

⁵⁸⁰ Weyers, AcP 182 (1982), S. 60 (73).

einen überschaubaren Satz von Regelungsproblemen begrenzt sind. Deshalb sei das Modell des BGB in seiner jetzigen Form geeignet, zu vorhersehbaren und konsensfähigen Entscheidungen im Einzelfall zu führen⁵⁸¹. Er räumt jedoch ein, dass die Verallgemeinerungsfähigkeit nicht für im Leistungsprogramm enthaltene Schutz-, Treu- und sonstige Nebenpflichten gilt⁵⁸².

Wie fällt diese zutreffende Beurteilung nun aus, wenn die ursprünglich als Nebenpflicht konzipierte Mitwirkung zu einem zentralen Bestandteil des Leistungsprogrammes wird, wenn die erfolgreiche Abwicklung des Vertrages mit der sorgsamten Erfüllung dieser „Nebenpflicht“ steht und fällt? Wie zuletzt bei Schneider in seinem Überblick über das IT-Vertragsrecht der letzten 20 Jahre anschaulich dargestellt, vollzieht sich die Softwareentwicklung in einem Pflichtengefüge, welches von der Verflechtung zweier Anpassungsleistungen geprägt ist⁵⁸³. Es zeigt sich das bereits dargestellte Element der Abhängigkeit der Vertragspartner voneinander. Zu den ohnehin bestehenden konzeptionellen Defiziten des Werkvertrages im Hinblick auf Langzeitverträge⁵⁸⁴ tritt hier also das Novum der engen Verbundenheit der Vertragspartner, die an eine gesellschaftsrechtliche Konstruktion denken lässt, ohne jedoch auf diesem Weg zufrieden stellende Lösungen anbieten zu können.

Das Fehlen einer zuverlässigen Verankerung im Gesetz wird durch die unzureichende Einarbeitung von Kooperationskonzepten in der untersuchten vertragsrechtlichen Praxis zu einem von der Rechtsordnung m.E. nicht mehr hinnehmbaren Gesamtbild komplettiert.

II. Die Lösung des Konfliktes zwischen Planung und Wirklichkeit

Planung und Wirklichkeit fallen regelmäßig auseinander. Dass die Vorstellung von einem Ergebnis nicht mit dem tatsächlichen Resultat übereinstimmt, ist keine Besonderheit von Softwareentwicklungsverträgen. Es offenbart sich bei ihnen jedoch in besonders prägnanter Form. Ein einmal

⁵⁸¹ Weyers, AcP 182 (1982), S. 61.

⁵⁸² Weyers, AcP 182 (1982), S. 77.

⁵⁸³ Schneider, IT-Vertragsrecht, in: CR 2005, S. 695 (S. 697f).

⁵⁸⁴ Weyers weist hier etwa auf die Unangemessenheit des jederzeitigen Kündigungsrechtes und der fehlenden Regelung für eine temporäre Leistungsstörung hin, AcP 182 (1982), S. 67ff.

errichtetes (Bau-) Fundament z.B. stellt eine unmittelbar und bei größeren Projekten eindrucksvoll wahrnehmbare Realität dar. Ebenso fundamentale Mengengerüste oder Datendefinitionen weisen diese sinnlich erfahrbare Eigenschaft nicht auf. Ein Bauherr, der sich mit einem Änderungsvorschlag trägt, wird sich häufig durch einmal geschaffene Fakten leiten lassen. Er wird kaum auf den Gedanken verfallen, ein etwa durch Änderungen der Bauplanung ermöglichtes, drittes Geschoss aufzusetzen, wenn der Architekt ihm erklärt, die Statik sei dafür nicht ausgelegt. Ein Softwarebesteller, der statt ursprünglich zwei nunmehr drei Produktgruppen in seinem Warenwirtschaftssystem verwalten will, kann vom Projektleiter kaum in ähnlich überzeugender Weise von seinem Vorhaben abgebracht werden. Selbst Änderungen grundlegender Art werden vor diesem Hintergrund nicht selten akzeptiert. Die nicht auf ein Dauerschuldverhältnis mit allfälligen Änderungen ausgerichteten in Betracht gezogenen Vertragstypen im deutschen Gesetz bieten kaum Hilfestellung. Um so größere Bedeutung fällt daher der vertraglichen Gestaltung zu, und um so erstaunlicher ist die unbefriedigende Ernte, die nach dieser Untersuchung hier konstatiert werden muss.

Das zuvor beschriebene Element der gemeinschaftlichen Projektabwicklung legt es nahe, notwendige und vor allem einseitig wünschenswerte Änderungen als gemeinsames Anliegen zu betrachten. Wirtschaftlich unabdingbar wird eine dementsprechende Interessenabstimmung dann, wenn es - wie in nahezu allen deutschen Vertragsmustern - eine Festpreisvereinbarung gibt. In diesem Fall ist das Softwarehaus darauf angewiesen, den Auftragsumfang sorgfältig mit der Vergütung abzustimmen, wenn es wirtschaftlich überleben will. Es ist daher in seinem ureigensten Interesse, Änderungen etwa im Hinblick auf eine Reduzierung des Leistungsspektrums selbst anstoßen und unkalkulierbare bzw. kalkulierbar nicht realisierbare Änderungen verhindern zu können. Eine entsprechende vertragliche Regelung ist nicht nur im Sinne des Softwarehauses, sondern auch im Sinne des Softwarebestellers. Hier sollte die Aussage aus dem zweiten Kapitel bedacht werden, wonach ein häufiger Grund für das Scheitern von Softwareprojekten darin zu sehen ist, dass das Softwarehaus nicht gelernt hat, oft oder wirkungsvoll genug „Nein“ zu sagen. Dies bedeutet nichts anderes als die Neigung der Softwarehäuser, auch schwer oder im

Ergebnis letztlich gar nicht erzielbare Ergebnisse als Vorgabe zu akzeptieren. Wenn diese Neigung auf eine Vertragsregelung trifft, die dem Softwarebesteller weitgehende und dem Softwarehaus allenfalls rudimentäre Rechte im Änderungsverfahren einräumt, so gerät das Projekt vom Regen in die Traufe. Ein ohnehin nur zögerlich die Anforderungen und Erwartungen „bremsendes“ Softwarehaus wird damit konfrontiert, Projekterweiterungen akzeptieren zu müssen.

Die natürliche Diskrepanz zwischen Planung und Wirklichkeit kann nicht durch einseitige Bestimmung gelöst werden. An ihre Stelle muss vielmehr eine gemeinsame Planung sich verändernder Wirklichkeit treten.

III. Interessenorientierte oder projektorientierte Vertragsgestaltung

Sowohl die deutsche wie die amerikanische Rechtsordnung sehen die Vertragsfreiheit als ein zentrales Element der Zivilrechtsverfassung an. Die Frage, was den Parteien im Rahmen der Gestaltung von Softwareentwicklungsverträgen *erlaubt* ist, stellt sich daher, wenn überhaupt, nur in extremen Fällen. Davon unabhängig ist allerdings die Frage, was aus Sicht der Beteiligten sinnvoll ist. Soll die Aushandlung des Vertrages dem freien Spiel der Kräfte überlassen, soll m.a.W. die Optimierung des Gesamtergebnisses also in der Optimierung der Eigeninteressen der Parteien gesucht werden? Wird dies mit einem Ja beantwortet, so treffen wir auf die bekannte marktwirtschaftliche Variante, die in eine interessenorientierte Vertragsgestaltung mündet.

Die Vertragsgestaltung kann jedoch auch projektorientiert erfolgen. In diesem Fall wird der Entwurf an dem Projektziel und daran orientiert, wie es am effektivsten zu erreichen ist.

Soweit die vergleichsweise geringe Stichprobe den Befund erlaubt, ist die Tendenz zu unausgewogenen Verträgen in den USA etwas ausgeprägter als in Deutschland. Der mit dem Vertragsentwurf befasste Jurist hat, je nach den Interessen und der Machtposition seines Auftraggebers, die Wahl zwischen einer softwarehaus- oder softwarebestellerfreundlichen Variante. Wählt er die erste, so schränkt er die Einstandspflichten des Softwarehauses

weitgehend ein, begrenzt den Auftragsumfang, der alleine und verbindlich vom Softwarebesteller zu definieren ist, und macht jede Zusatzleistung von einer Zustimmung und Vergütungspflicht abhängig. Wählt er die zweite, so werden die Estandspflichten möglichst weitgehend gefasst und der Auftragsumfang einschließlich wahrscheinlicher Änderungen wird - bei einem Festpreis - im Projektverlauf durch den Softwarebesteller bestimmt.

Nach den Erkenntnissen des zweiten Kapitels sollte die Zieldefinition auf der Grundlage kontinuierlicher Abstimmung der Vertragspartner sorgfältig und mit möglichst geringem Änderungspotential erarbeitet und im Verlauf des Projektes festgelegt werden.

Mit keiner der interessenorientierten Varianten ist diese Vorgehensweise vereinbar. In der ersten findet die Zieldefinition zu einem ungünstigen (zu frühen) Zeitpunkt statt, in der zweiten werden die Änderungen gar nicht oder nicht effektiv eingegrenzt. In beiden wird dem Kooperationserfordernis keine Rechnung getragen. Diese Form der Vertragsgestaltung ist daher nicht nur unausgewogen, sondern auch ineffektiv, d.h. ungeeignet, den Projekterfolg zu fördern.

Unausgewogene Verträge bedeuten ein hohes Projektrisiko. Sie spiegeln ungleiche Machtpositionen oder ein - vermeintliches - Verhandlungsgeschick einer Vertragspartei wider, nicht jedoch die aus informationstechnischer Sicht erforderlichen Rahmenbedingungen. Die zusätzliche Ineffektivität solcher Verträge erhöht dieses Projektrisiko weiter. Ein vermeintlicher Verhandlungserfolg bzw. die Durchsetzung der eigenen Position kann also leicht in ein Scheitern des Projektes führen, welches zwar möglicherweise durch Haftungsregelungen abgesichert ist, aber gleichwohl ein unbefriedigendes Ergebnis langer beidseitiger Bemühungen darstellt.

Aus Sicht der Projektleitung ist ein „guter Vertrag“ ein solcher, der den Projekterfolg begünstigt und ihm nicht vermeidbare Hindernisse entgegenstellt. Eine projektorientierte Vertragsgestaltung ist aus diesem Blickwinkel unzweifelhaft vorzugswürdig.

B. Ergebnis

Die Einordnung eines deutschen und eines amerikanischen Softwareentwicklungsvertrages in das System der Rechtsordnungen ist nicht gefestigt. Im Gegenteil: Die deutsche Schuldrechtsreform hat in Fachkreisen zu einer erheblichen Verunsicherung beigetragen⁵⁸⁵. Im amerikanischen und deutschen Recht ist nicht vorhersehbar, ob bzw. welche Normen vor Gericht angewendet bzw. bei der Vertragsgestaltung als Referenz betrachtet werden müssen. Die Leistungsbestimmung kann dabei auf deutscher Seite maßgeblich von der Einordnung geprägt sein, insbesondere wenn die außerhalb der Hauptströmungen diskutierten Modelle (Lizenzvertrag, Gesellschaftsvertrag, Dienstvertrag) wiederbelebt werden⁵⁸⁶. Auch in den USA übt die (Nicht-) Anwendbarkeit des U.C.C. z.B. hinsichtlich der Einstandspflicht erheblichen Einfluss auf die Entscheidungsfindung aus. Im Ergebnis kann in beiden Rechtsordnungen bzgl. der Beurteilung von Softwareentwicklungsverträgen von Rechtssicherheit keine Rede sein.

Rechtssicherheit im Sinne von Vorhersehbarkeit der rechtlichen Folgen des Handelns repräsentiert ein tragendes Prinzip des deutschen und des amerikanischen Rechts. Während sich das Common Law dabei im Vertragsrecht in erster Linie auf ein „*system of jurisprudence founded on the immutable principles of justice*“ stützt⁵⁸⁷, sucht der deutsche Gesetzgeber dieses Prinzip durch Kodifikation zu verwirklichen. Dass dies nicht immer gelingt, versteht sich von selbst⁵⁸⁸. Die Möglichkeit der Kodifikation bietet jedoch den auch von Vertretern des Common Law anerkannten Vorteil, „*ex ante rules*“ zu schaffen⁵⁸⁹, d.h. Reformen mit abschließender Wirkung für die

⁵⁸⁵ Redeker weist dementsprechend im Zusammenhang mit der Schuldrechtsreform darauf hin, dass der „Gesetzgeber Probleme auch schaffen kann, wenn er die Spezifikation besonderer (bei Software aber mittlerweile weit verbreiteter) technischer Produkte nicht hinreichend berücksichtigt“. In: CR 2005, S. 700 (705).

⁵⁸⁶ Hier sei beispielhaft nur auf den provokativen Titel des Aufsatzes von Schneider verwiesen: „Softwareerstellung und Softwareanpassung – Wo bleibt der Dienstvertrag? In: CR 2003, S. 317.

⁵⁸⁷ Maryland's Chief Judge Chase, zitiert nach Madden, Start; The vital Common Law: Its role in a statutory age, 18 U. Ark. Little Rock L.J. 555.

⁵⁸⁸ Anschaulich bei Salmond im Anschluss an eine deutliche Stellungnahme zu Gunsten von Kodifikationslösungen im englischen Recht formuliert: „*In the most carefully prepared of codes subtle ambiguities will come to light, real or apparent inconsistencies will become manifest, and omissions will reveal themselves. No legislative skill can effectually anticipate the complexity and variety of the facts.*“ In: Salmond on Jurisprudence, 12th ed., London, 1966, S. 131.

⁵⁸⁹ Madden, a.a.O., Fn. 562.

Vergangenheit und Zukunft zu fassen⁵⁹⁰. Durch eine gesetzliche (Neu-)Regelung könnte also Abhilfe geschaffen werden. Andererseits hat das BGB Herausforderungen unterschiedlichster Natur bestanden und die Besinnung auf eine generalisierende Betrachtungsweise wird mit guten Gründen immer noch als geeignete Antwort auf die Vielgestaltigkeit der Dienstleistungsverträge verstanden⁵⁹¹.

In der angloamerikanischen Diskussion zur Frage, ob eine richterrechtliche Lösung (*precedent*) der gesetzlichen (*statute*) vorzuziehen sei, wird als Vorteil der ersteren deren Flexibilität und Fähigkeit zu Wachstum und Anpassung hervorgehoben⁵⁹². Das Recht ändere sich dann im Wege der Evolution und nicht durch Revolution⁵⁹³. Diese schrittweise Entwicklung ermögliche es zudem, auf den richtigen Moment für eine neue weichenstellende Entscheidung zu warten⁵⁹⁴. Die beste Lösung werde durch fallbezogene Antworten (der Richter) gefunden⁵⁹⁵.

Aus deutscher Sicht dürfen an dieser Stelle zwei Aspekte nicht unbeachtet bleiben. Zum einen hat die Rechtsprechung vor allem der Obergerichte zwar eine faktisch bedeutsame Wirkung, sie bleibt jedoch v.a. methodisch deutlich hinter der amerikanischen Entscheidungen zurück. Das OLG München entwickelt seine Begründung nicht auf Grundlage eines Urteils des OLG Köln (oder auch eines eigenen Urteils), sondern zieht allenfalls die dort verwendeten Argumente zur Stützung der eigenen Argumentation heran.

⁵⁹⁰ Salmond führt dazu aus, „*the first virtue of legislation lies in its abrogative power. It is not merely a source of new law, but is equally effective in abolishing that which already exists (..) in a strict system of binding precedent its operation is irreversible. What it does, it does once for all.*“ A.a.O., S. 125.

⁵⁹¹ Weyers, AcP 182 (1982), S. 77.

⁵⁹² „... *case law, with all its imperfection has at least this merit, that it remains in living contact with the reason and justice of the matter, and draws from this source a flexibility and a power of growth and adaption which are to much wanting in a the litera scripta of enacted law.*“ Salmond, a.a.O., S. 129.

⁵⁹³ Levine, Howard, Hugh R. Jones Lecture at Albany Law School; 67 Alb. L. Rev. S. 1 (21).

⁵⁹⁴ „*The process of judging I have described, of course, is not perfect. Mistakes will be made. The social realities may be wrongly perceived. Unforeseen, embarrassing consequences may occur. But the commitment to incremental development of the law - to waiting for just the right moment when a broader rule may be articulated and justified as underlying prior rulings - also, I believe, best enables correcting or at least limiting the damage of mistakes.*“ Jones, a.a.O., S. 25.

⁵⁹⁵ „*Law always reflects a community's values, and American contract law reflects the continuing conflict in our society between individual freedom and public control. This conflict is unavoidable in a liberal democracy like ours and grand theories can do little to resolve it. The best the law can do is fashion reasonable compromises on a case-by-case basis.*“ Mark L. Movsesian, Two Cheers for freedom of contract: The Fall and Rise of Freedom of Contract. In: 23 Cardozo L. Rev. S. 1529 (1547f).

Zum anderen ist die Kontinuität der Rechtsprechung (Einordnung von Softwareentwicklungsverträgen als Werkvertrag) durch die Schuldrechtsreform, eine „Revolution“ im soeben genannten Sinne, in Frage gestellt.

Unabhängig davon bedürfen weitere Aspekte der Berücksichtigung.

Die Vielfalt der oben (unter Technik und Recht) aufgeführten Abweichungen typischer Elemente eines Softwareentwicklungsvertrages vom klassischen Bild der im BGB vorgefundener Regelungsmodelle setzt dessen Begriffswelt erheblichen Spannungen aus. Diese Spannung kann, wie sich am Beispiel der Einordnung von Software zeigen lässt, bisweilen nur mit durchaus fragwürdigen Konstruktionen gelöst werden.

Unter der Prämisse, dass „nur ganz wenige, wichtige Lebensgebiete, wie etwa die Vertragsbeziehungen im Bau- und Transportwesen“ erfassbar und getrennt regelbar sind⁵⁹⁶, fragt sich, ob Software mittlerweile dieser Gruppe zugerechnet werden muss.

Die Antwort ist m.E. eindeutig. Nicht ohne Grund wird heute von einem „Informationszeitalter“ gesprochen. An der wirtschaftlichen Bedeutung der Softwareindustrie besteht ohnehin kein ernsthafter Zweifel⁵⁹⁷. Der amerikanische Gesetzgeber war bereits Ende der neunziger Jahre der Auffassung, Information und Software verdiene ein eigenes Gesetz⁵⁹⁸. Hält man sich weiter vor Augen, dass eine sehr spezielle - wenn auch zweifellos sehr bedeutsame - Regelungsmaterie, die Reise, ihren Platz im Kanon der Modelle des besonderen Schuldrechts gefunden hat, so fragt sich, mit welchen Gründen der Software ein Anspruch auf eine gesetzliche Grundlage verweigert werden kann.

Die Diskussion um Softwareverträge ist auch keineswegs eine Eintagsfliege. Sie wird mit unterschiedlichen Vorzeichen bereits seit über zwanzig Jahren

⁵⁹⁶ Weyers, AcP 182 (1982), S. 61.

⁵⁹⁷ S.o. Kapitel 1 B. (S. 15ff).

⁵⁹⁸ S.o. Kapitel 3 A. (S. 38) und C. III. (S. 49).

geführt⁵⁹⁹ und ist durch die Schuldrechtsreform neu belebt worden. Ein Ende ist nicht in Sicht, zumal die Vielfalt der im Zusammenhang mit Software auftauchenden Sachverhalte regelmäßig durch neue Varianten bereichert wird⁶⁰⁰.

Darüber hinaus zeigt der Befund aus den untersuchten Mustern und Verträgen hinsichtlich bedeutsamer Einzelfragen, dass es teilweise nur schwer oder gar nicht gelingt, Freiräume und Defizite im rechtlichen Rahmen auszugleichen.

Schließlich besteht ein unabweisbares internationales Interesse an einer Lösung. Software wird weltweit produziert, angepasst, erweitert und mit Hilfe des Internet auch vertrieben, gewartet und durch andere Dienstleistungen (wie z.B. Sicherung (Backup)) begleitet. Für die Bewältigung der damit verbundenen Rechtsfragen wäre daher eine Basis, die grundlegende Fragen einer Antwort zumindest näher bringt, von allgemeinem Interesse.

Die Vielzahl der Projekte, die teilweise ohne präzise Leistungsbeschreibung oder ohne geeignete vertragliche Grundlage zu Recht und letztendlich vor Gericht kommen (müssen), erfordert die Tätigkeit des Gesetzgebers. Die Rechtsprechung hat sich in jahrzehntelanger Praxis auf eine Zuordnung der Softwareentwicklungsverträge zum Werkvertragsrecht versteift. Von ihr kann eine nachhaltige Lösung kaum erwartet werden. Auf Ebene der Softwareverbände unter federführender Beteiligung der öffentlichen Auftraggeber sind zwar allgemeine Bedingungen auch für den Bereich der Planung und Erstellung von Software geschaffen worden (die BVB bzw. EVB-IT), doch ist der bereits Anfang der (19)70er Jahre begonnene Entwicklungsprozeß noch nicht zum Abschluss gekommen und es ist diesen allgemeinen Regeln bisher nicht gelungen, einen der VOB auch nur annähernd vergleichbaren Stellenwert zu erlangen.

Oben (Kapitel 8 A. II) wurden drei Möglichkeiten für eine Reform durch den Gesetzgeber angedeutet: (1) Die ausdrückliche Zuweisung von

⁵⁹⁹ Zusammenfassende Darstellungen finden sich bei Redeker, CR 2005, S. 695 und Schneider, CR 2005 S. 700, die anlässlich des zwanzigjährigen Jubiläums der Zeitschrift CR auf zahlreiche, immer noch bzw. wieder ungelöste Probleme hinweisen.

⁶⁰⁰ Stellvertretend können hier Application Service Providing (dazu Nolting, Roger-David\ Nolte, Bernd; in: WiSt 2004, 431-435) und Open Source Software (dazu Spindler, Gerald (Hrsg.) Rechtsfragen bei Open Source, Köln, 2004) genannt werden.

Softwareentwicklungsverträgen zu einem bestimmten, vorhandenen Vertragstypus, (2) die Schaffung eines neuen Vertragstypus oder (3) die Einführung eines Computer- und Informationsgesetzes.

Die erste Möglichkeit würde nach den Ergebnissen dieser Untersuchung sehr wahrscheinlich zur Anwendung des Werkvertragsrechts führen. Die §§ 631 ff BGB sind jedoch als effektive Rechtsgrundlage für Softwareentwicklungsverträge ungeeignet. Effektiv ist eine Rechtsgrundlage dann, wenn sie bezüglich der zentralen Fragen, die mit einem Sachverhalt verknüpft sind, Antworten für die Parteien bereithält. Dies ist jedoch nicht der Fall.

Zunächst existiert kein klares Leistungsprofil. Die Umschreibung als „Herstellung eines versprochenen Werkes“ eröffnet mehr Fragen, als sie beantwortet. Die Mitwirkung des Bestellers, die in Softwareentwicklungsverträgen eine unabdingbare Voraussetzung für den Projekterfolg darstellt, ist in § 642 BGB nur rudimentär geregelt. Die Konsequenz der §§ 643, 645 Abs.1 S.2 BGB stellt zudem selten eine sinnvolle Wahl des Softwarehauses dar. Die zentrale Bedeutung der Mitwirkung zeigt sich besonders in der ersten Phase des Projektes, wo die zunehmend präzisere Bestimmung der Leistung aus dem intensiven Zusammenwirken der Vertragsparteien hervorgeht. Eine Mitwirkung des Bestellers ist in § 643 BGB (nur) als Nebenpflicht vorgesehen, tatsächlich bildet sie in Softwareentwicklungsverträgen jedoch die Grundlage für die Definition und Ausführung der zu erbringenden Leistung.

Der Projekterfolg, im rechtlichen Sinne also die vertragsgemäße Erfüllung, wird am Ende durch den Vergleich von Leistungsbestimmung und Leistungsergebnis gemessen. Der Komplexität der Leistungsbestimmung entsprechend, fällt in der Softwareentwicklung auch die Leistungskontrolle aus. Die Abnahme findet regelmäßig in verschiedenen Stufen statt und ist nicht selten Ausgangspunkt für weitere wünschenswerte oder notwendige Arbeiten des Softwarehauses. Für ein solches Szenario ist die Abnahmeregung des § 640 BGB nicht ausgelegt. Das Werkvertragsrecht versagt demnach, wenn es von den Parteien bezüglich der zu Beginn, bei allfälligen Änderungen oder bei Abschluss des Leistungen anfallenden

Fragen untersucht wird. Die erste der oben genannten Möglichkeiten scheidet daher aus.

Es verbleiben die Optionen, einen neuen Vertragstyp oder ein vollständig neues (Softwarevertrags-) Gesetz ins Leben zu rufen.

Ein Gesetz, das Softwareverträge in den Mittelpunkt stellt, könnte sich nicht auf die Softwareentwicklung beschränken. Zahlreiche andere Konstellationen aus der Praxis müßten mit bedacht werden⁶⁰¹. Es fehlt jedoch an den maßgeblichen Kriterien, um eine sinnvolle Abgrenzung der rechtlich zu regelnden Thematik vornehmen zu können, was nicht zuletzt auf den Einfluß des Internet zurückzuführen ist. Hier entstehen beständig neue Geschäftsmodelle mit entsprechendem Regelungsbedarf, die meist unmittelbar mit dem Einsatz und der Entwicklung von Software verknüpft sind. Die Zeit für einen solchen „großen Wurf“, der die Vielfältigkeit des Umganges mit Software zufriedenstellend erfaßt, ist m.E. noch nicht gekommen.

Demgegenüber stellt die Softwareentwicklung eine vergleichsweise etablierte Projektform dar, deren problematische Gesichtspunkte an zahllosen praktischen Beispielen auch in der Rechtsprechung nachvollzogen werden können. Diese Untersuchung hat dabei gezeigt, dass sich ein Kernbereich dieser Gesichtspunkte durchaus umreißen läßt. Es ist der bereits eingangs⁶⁰² erwähnte von Austausch und kontinuierlicher Prüfung geprägte Prozess stufenweise konkretisierenden Vorgehens.

Im Ergebnis steht für den Gesetzgeber daher die Aufgabe an, einen Vertragstyp *Softwareentwicklung* zu schaffen. Dieser sollte sich an der Form orientieren, in der das Zusammenwirken der Vertragsparteien bei der Softwareentwicklung üblicherweise erfolgt (Verflechtungsgedanke). Er muß darüber hinaus die als Prozeß zu begreifende Entstehung einer Zieldefinition und die aus der Flüchtigkeit des bearbeiteten Mediums

⁶⁰¹Als Beispiele sind hier zu nennen: Planung, Beratung, Überlassung und Pflege im Hinblick auf Software oder EDV (Software und Hardware) sowie die aus der Praxis entstandenen, bzgl. ihrer Einordnung noch nicht klar konturierten Outsourcing-, Hosting-, Application Service Providing-, Open-Source und Datenbankverträge, die jeweils ein Bündel von Leistungen enthalten können.

⁶⁰²In der Einleitung, S. 2.

resultierende Änderungsanfälligkeit berücksichtigen (Entwicklungsgedanke). Auf dieser Grundlage könnte ein neuer Titel im besonderen Schuldrecht entstehen, der auch als Modell für ähnlich gelagerte Konstellationen, z.B. für Projektverträge, in Frage kommt.

Ein bedeutender Teil der vorliegenden Arbeit hat sich mit Änderungen, mit ihrer vorhersehbaren Unvorhersehbarkeit, mit ihren weitreichenden Auswirkungen und mit der einem Paradoxon angenäherten Spannung zwischen Wunsch und Wirklichkeit befasst. Diese Spannung war Ausgangspunkt und häufig Triebfeder für den Versuch, die Thematik zu erforschen. Sie mag sich auch als Triebfeder für das Wagnis eines größeren Entwurfes erweisen, zu dem der Autor hofft beigetragen zu haben. Als Grundlage für einen solchen Entwurf können die folgenden Bestimmungen dienen:

Einzelne Schuldverhältnisse

Titel 8a. Softwareentwicklungsvertrag

§§ 630a-630d

§ 630a Leistungsbestimmung. (1) Der Besteller kann die Leistung vom Unternehmer nach Maßgabe einer Leistungsbeschreibung verlangen, soweit nicht etwas Abweichendes vereinbart ist.

(2) Soweit die Leistungspflicht unbestimmbar ist, kann der Besteller vom Unternehmer nur die bestimmbare Leistung verlangen. Der Unternehmer kann vom Besteller nur die Vergütung für die bestimmbare Leistung verlangen.

§ 630b Änderung der Leistungsbestimmung. Der Unternehmer kann die Erbringung seiner Leistung, der Besteller die Zahlung der Vergütung von einer Änderung der Leistungsbestimmung abhängig machen, soweit ein Festhalten an der vereinbarten Leistung unter Berücksichtigung aller Umstände des Einzelfalles nicht zumutbar ist.

§ 630c Mitwirkungspflichten. Der Unternehmer kann vom Besteller die zum Leistungserfolg erforderlichen Mitwirkungshandlungen verlangen. Der Unternehmer kann sich auf die Mitwirkungspflicht des Bestellers nicht berufen, soweit der Besteller sie nicht ohne Beratung oder sonstige Handlung des Unternehmers erbringen kann.

§ 630d Leistungsmaßstab. Der Unternehmer kann die Vergütung verlangen, wenn er seine Leistung gemäß Leistungsbeschreibung im wesentlichen erbracht hat. Für Teilleistungen gilt dies, soweit eine Vergütung für Teilleistungen vereinbart ist.

Literaturverzeichnis

A

- Anwaltskommentar BGB Band 2. Schuldrecht, Teilband 1: §§ 241 bis 610,
Mönchengladbach 2005 (zitiert: AnwK/Bearbeiter)
- Band 2 Schuldrecht, Teilband 2: §§ 611 bis 853,
Mönchengladbach 2005 (zitiert: AnwK/Bearbeiter)

B

- Balzert, Helmut Lehrbuch Grundlagen der Informatik, 2. Aufl.,
München 2005
- Bamberger, Heinz Georg/Roth, Herbert (Hrsg.)
 Kommentar zum Bürgerlichen Gesetzbuch, München
2003 (zitiert: Bamberger/Roth / Bearbeiter)
- Bartl, Harald Rechtliche Problematik der Softwareverträge, BB
1988, 2122
- Bauer, Indes M./Witzel, Michaela
 Individualsoftwareerstellung - § 651 und die
Neugestaltung des Abnahmeverfahrens, ITRB 2003,
S. 62
- Baumbach, Adolf/Hopt, Klaus J.
 Handelsgesetzbuch, 31. Aufl., München 2003
- Blumenwitz, Dieter Einführung in das anglo-amerikanische Recht, 7. Aufl.,
München 2003
- Bormuth, Ulrich/Schmitz, Karl
 Beteiligung der Anwender an der Softwareentwicklung,
AiB 1993, S. 31
- Braun, Bernd Überlassung von Computerprogrammen an den
Anwender als Werkvertrag, BB 1992, S. 154
- Brockhaus Computer und Informationstechnologie, Mannheim
Leibzig 2003
- Brooks, Frederick P. EDV-Projekte in den Griff bekommen, Köln 1987
- ders. Management von Informatik-Projekten, München Wien
1997
- ders. Vom Mythos des Mann-Monats, Bonn 1987
- Bröhl, Adolf-Peter/Dröschel
 Das V-Modell, 2. Aufl., München 1995

C

- Calamari, John D./Perillo, Joseph M.
 The law of contracts, 4 ed., St. Paul, Minnesota 1998
- California Transaction Forms Westlaw Database § 9:59, 2003 (Slater, Amy)

Literaturverzeichnis

- Carter, J. W. Article 2B: International Perspectives; Journal of Contract Law 1999, S. 54
- Corbin, Arthur Linton Corbin on Contracts, Revised Edition, St. Paul 1993
- CORI Datenbank des Contracting and Organizations Research Institute der University of Missouri, Columbia (cori.missouri.edu/search)
- Corvaglia, Stefano Das einheitliche UN-Kaufrecht, Bern 1998
- Cox, Trevor Chaos versus uniformity: the divergent views of software in the International Community; Business Law International 2000, 359

D

- Dauner-Lieb, Barbara/Heidel, Thomas/Ring, Gerhard (Hrsg.)
Anwaltskommentar BGB - Band 2. Schuldrecht,
Teilband 1: §§ 241 bis 610, Mönchengladbach 2005
(zitiert: AnwK/Bearbeiter)
- Anwaltskommentar BGB - Band 2 Schuldrecht,
Teilband 2: §§ 611 bis 853, Mönchengladbach 2005
(zitiert: AnwK/Bearbeiter)
- De Marco, Tom The Deadline; New York 1997
- Diedrich, Kay Typisierung von Softwareverträgen nach der Schuldrechtsreform, CR 2002, S. 473
- Dubischar, Roland Grundbegriffe des Rechts, Stuttgart Berlin Köln Mainz 1968
- Dumke, Rainer Software Engineering, 3. Aufl., Braunschweig Wiesbaden 2001

E

- Eckert, Jörn AcP 2003, 863ff
- EDGAR Datenbank der US Securities and Exchange Commission (<http://www.sec.gov/edgar.shtml>)
- Endler, Maximilian/Daub, Jan Internationale Softwareüberlassung und UN-Kaufrecht, CR 1993, S. 601
- Erman Bürgerliches Gesetzbuch Handkommentar, 11. Aufl., Köln, 2004 (zitiert: Erman/Bearbeiter)
- Eschbach, Andreas Software nach Maß, Haar 1992
- Esser, Josef/Weyers, Hans-Leo Schuldrecht Band II Besonderer Teil, 8. Aufl., Heidelberg 1998
- Esser, Josef/Schmidt, Eike Schuldrecht Band I Allgemeiner Teil Teilband 1, 8. Aufl., Heidelberg 1995

Literaturverzeichnis

Etzel, Hans Joachim/Heilmann, Heidi/Richter, Reinhard (Hrsg.);
IT-Projektmanagement, Heidelberg 2000

F

Feretti, Vittorio Wörterbuch der Datentechnik, Berlin Heidelberg New
York 1996

Fischer, Joachim/Herold, Werner/Dangelmaier, Wilhelm/Nastansky, Ludwig/
Suhl, Leena Bausteine der Wirtschaftsinformatik, 3. Aufl., Berlin
2002

Fisher, David T. Myths and Methods: a guide to software productivity,
New York 1991

Flechtner, Harry M. Substantial Revisions to U.S. Domestic Sales Law,
IHR, 2004

Florida Jur Forms Legal and Business Database § 20B:1, 2003

Flume, Werner Vom Beruf unserer Zeit für Gesetzgebung, in: ZIP
2000, S. 1427

Franz, Wolfgang Techniken der modernen Software-Entwicklung,
München 1991

Frühauf, Karol/Ludewig, Joachen/Sandmayr, Helmut
Software-Prüfung, Stuttgart 1991

G

Grupp, Bruno EDV-Pflichtenheft zur Hardware- und
Softwareauswahl, 2. Aufl., Köln 1991

Grieser, Franz/Irlbeck, Thomas
Computer Lexikon, 2. Aufl., München 1995

H

Hansen, Hans Robert/Neumann, Gustaf
Wirtschaftsinformatik I, Grundlagen betrieblicher
Informationsverarbeitung, 8. Aufl., Stuttgart 2001

Hay, Peter U.S.-Amerikanisches Recht, 2. Aufl., München 2002

Heinrich, Lutz J. Systemplanung, 6. Aufl., München Wien 1994

Heppner, Henning Softwareerstellungsverträge, Diss., Köln 1997

Herr, Robert Anm. zu LG Karlsruhe, U. v. 13.05.1991, 10 O 458/89,
CR 1992, S. 342

Hilty, Reto Der Softwarevertrag – Ein Blick in die Zukunft, MMR
2003, S. 3

Hoeren, Thomas Softwareüberlassung als Sachkauf, Diss., Münster
1989

I

Ihde, Thomas Das Pflichtenheft beim Softwareerstellungsvertrag, CR
1999, S. 409

J

Jauernig, Othmar (Hrsg.) BGB Kommentar, 11. Aufl., München 2004 (zitiert: Jauernig/Bearbeiter)

K

Karger, Michael "Kooperation" bei komplexer Softwareentwicklung. Konsequenzen unzureichender vertraglicher Regelung der Zusammenarbeit, ITRB 2004, S. 208

Kargl, Herbert Management und Controlling von IT-Projekten, München Wien 2000

Koch, Frank Kaufrechtliche Vorgaben für Verträge zur Softwareerstellung, ITRB 2002, S. 297

ders. Computer-Vertragsrecht, 6. Aufl., Freiburg 2002

Kottof, Jost Softwareerstellungsverträge nach der Schuldrechtsreform, K&R 2002, S. 105

Köhler, Helmut Rechtsfragen zum Softwarevertrag, CR 1987, S. 827

Krauß, Wolfgang Die Auswirkungen der Schuldrechtsreform auf Softwareverträge, Diss., Münster 2004

L

Levine, Howard A. Hugh R. Jones Lecture at Albany Law School; 67 Albany Law Rev. (Alb. L. Rev.) S. 1

Love, Tom Tücke des Projekts (Object Lessons), Hannover 1994

M

Macneil, Ian R./Gudel, Paul J. Contracts, 3. Aufl., New York 2001

Madden, Start The vital Common Law: Its role in a statutory age, 18 U. Ark. Little Rock L.J. 555

Mall, Rajib Fundamentals of Software Engineering, New Delhi 2003

Malzer, Hans-Michael Der Softwarevertrag, Diss., Köln, 1991

Marly, Jochen Softwareüberlassungsverträge, 4. Aufl., München 2004

Maymon, Gilbert W./Cave, William C. Leitfaden des Software-Projektmanagements, Wiesbaden 1988

McManus, John/Wood-Harper, Trevor Information Systems Project Management, Delhi 2004

Merryman, John Henry The Civil Law Tradition, 2. Aufl., Stanford California 1985

- ders. Western European and Latin American Legal Systems, Charlottesville Virginia 1978
- Movsesian, Mark L. Two Cheers for freedom of contract: The Fall and Rise of Freedom of Contract. In: 23 Cardozo L. Rev., S. 1529
- Münchener Kommentar zum Bürgerlichen Gesetzbuch
Bd. 1 Allgemeiner Teil, München 4. Aufl. 2001

Bd. 4. Schuldrecht Besonderer Teil, 4. Aufl. 2005
(zitiert: MüKo/Bearbeiter)
- Münchener Kommentar zum Handelsgesetzbuch
Bd. 6 §§ 373-406, CISG, München 2004 (zitiert MüKo-HGB/Bearbeiter)
- Müller-Hengstenberg, Claus Dieter
Der Vertrags als Mittel des Risikomanagements – Ein Plädoyer für die dynamische Projektbegleitung im Vertrag, CR 2005, S. 385
- ders. Vertragstypologie der Computersoftwareverträge, CR 2004, S. 161
- ders. BVB-EVB-IT-Computersoftware, 6. Aufl., Berlin 2003
- N**
- Nichols Cyclopedia of Legal Forms Annotated, Westlaw Database, § 2.3191, 2002 (Savage, Diane W.)
- Nicklisch, Fritz/Weick, Günter
Verdingungsordnung für Bauleistungen, 3. Aufl. München 2001
- Nicklisch, Fritz
Kooperation bei Projektverträgen am Beispiel von Softwareentwicklungsverträgen, FS Traub, 1994, S. 305
- Nolting, Roger-David/Nolte, Bernd
Application Service Providing, WiSt 2004, S. 431
- Norvis, Mark
Survival in the Software Jungle, Boston 1995
- P**
- Palandt, Otto (Begr.)
Bürgerliches Gesetzbuch, 64. Aufl. München 2005
(zitiert: Palandt/Bearbeiter)
- Plath, Kai-Uwe
Abnahme bei Individualsoftwareverträgen, ITRB 2002, S. 98
- Primak, Scott L.
Computer Software: Should the U.N. Convention on contracts for the international sale of goods apply?; Computer Law Journal 1991, S. 197

R

- Raysman, Richard/Brown, Peter
Computer Law: Drafting and Negotiating Forms and Agreements, Lsbl. New York 2003
- Redeker, Helmut
IT-Recht in der Praxis, 3. Aufl., München 2003
- ders.
Softwareerstellung im neuen Schuldrecht, ITRB 2002, S. 202
- ders.
Softwareerstellung und § 651 BGB, CR 2003, S. 88
- ders.
Von Dauerbrennern und neuen Entwicklungen im Recht der Leistungsstörungen. Wie sich technische und gesetzgeberische Innovationen auf juristische Diskussionen ausgewirkt haben, CR 2005, S. 700
- Rehmann, Wolfgang
Herstellung von Individualsoftware – Folgen der Verletzung von Mitwirkungspflichten, CR 1989, S. 961
- Rupp, Chris
Requirements-Engineering und -Management, 2. Aufl. München, Wien 2002

S

- Salmond, John
Salmond on Jurisprudence, Twelfth Ed., London 1966
- Schaub, Beate
Das Pflichtenheft im Spiegel der Rechtsprechung, CR 1993, S. 329
- Schlechtriem, Peter (Hrsg.)
Kommentar zum Einheitlichen UN-Kaufrecht, 3. Aufl., München 2000 (zitiert: Schlechtriem / Bearbeiter)
- Schlosser, Hans
Grundzüge der Neueren Privatrechtsgeschichte, 9. Aufl., Heidelberg 2001
- Schmid, Joachim
Probleme der Rechtsnaturbestimmung moderner Vertragstypen am Beispiel des „Back-Up-Vertrages“, CR 1994, S. 513
- Schmidl, Michael
Softwareerstellung und § 651 BGB – ein Versöhnungsversuch, MMR 2004, S. 590
- Schneider, Jochen
Handbuch des EDV-Rechts, 3. Aufl., Köln 2003
- ders.
Softwareerstellung und Softwareanpassung – Wo bleibt der Dienstvertrag?, CR 2003, S. 317
- ders.
IT-Vertragsrecht, CR 2005, S. 695
- Schneider, Jochen/Bischof, Elke
Das neue Recht für Softwareerstellung/-anpassung, ITRB 2002, S. 273
- Schröder, Georg F.
Softwareverträge, München 2002
- Schubert, Michael von/Haase, Marina
Konfliktmanagement in IT-Projekten, ZKM 2003, S. 251
- Schütze, Rolf A. (Hrsg.)
Münchener Vertragshandbuch, 5. Aufl. München 2004 (Zitiert: Münchener Vertragshandbuch / Bearbeiter)

- Schweinoch, Martin/Roas, Rudolf
Paradigmenwechsel für Projekte: Vertragstypologie der Neuerstellung von Individualsoftware, CR 2004, S. 326
- Schweizerische Vereinigung für Datenverarbeitung (Hrsg.)
EDV-Pflichtenhefte, 2. Aufl., Bern und Stuttgart 1984
- Schwickert, Axel
Web Site Engineering, Stuttgart Leipzig Wiesbaden 2001
- Scott, Michael D.
Scott on Computer Law, 2. Aufl., New York 2003
- Seffer, Adi/Horter, Carsten
Nebenleistungspflichten des Erstellers von Individualsoftware, ITRB 2005, S. 169
- Sick, Ulrich
Verträge im Projekt- und Systemgeschäft, 2. Aufl., Heidelberg 2004
- Simon, David F.
Computer Law Handbook, Philadelphia 1990
- Slongo, Doris
Der Softwareerstellungsvertrag, Diss., Zürich 1991
- Sobola, Sabine/Dobmeier, Gerhard (Hrsg.)
Software- und Arbeitsverträge für die IT-Branche, Berlin 2003 (zitiert: Sobola/Dobmeier/Bearbeiter)
- Söbbing, Thomas
Die Bedeutung der ITIL für die Vertragsgestaltung, ITRB 2005, S. 97
- Soergel, Hs. Th. (Begr.)
Bürgerliches Gesetzbuch, Bd. 2 Allgemeiner Teil §§ 104-240, 13. Aufl., Stuttgart 1999 (zitiert Soergel/Bearbeiter)
- ders.
Bd. 13 CISG, 13. Aufl. Stuttgart 2000
- Spindler, Gerald (Hrsg.)
Rechtsfragen bei Open Source, Köln 2004
- Spitta, Thorsten
Softwareentwicklung, Berlin 2003
- ders.
Scriptum Informationsmanagement, S. 5, abrufbar (zuletzt 12.12.05) unter <http://www.wiwi.uni-bielefeld.de/~spitta/download/IM3.pdf>
- Statistical Abstract of the US 2002, 122nd Edition, Issued by the Department of Commerce, Dec. 2002
- Statistisches Jahrbuch 2005 für das Ausland (SJB-A 2005), Wiesbaden 2005
- Statistisches Jahrbuch 2005 für die Bundesrepublik Deutschland (SJB-D 2005), Wiesbaden 2005
- Staudinger, Julius von (Begr.)
Einl. zum BGB, §§ 1-12, Verschollenheitsgesetz, 12. Aufl., Berlin 1995
- ders.
BGB Wiener UN-Kaufrecht, 12. Aufl., Berlin 2005 (zitiert: Staudinger/Bearbeiter)

T

- Thaller, Georg Erwin Software- und Systementwicklung, Hannover 2002
- Thewalt, Stephan Der Softwareerstellungsvertrag nach der
Schuldrechtsreform, Diss., München 2004

U

- U.C.C 2003 Edition, Philadelphia Chicago 2003

V

- Volle, Peter Rechtliche Einordnung der EDV-Systemverträge, CR
1996, S. 139

W

- Weick, Günter Vereinbarte Standardbedingungen im deutschen und
englischen Bauvertragsrecht, München 1977
- Weyers, Hans-Leo (Hrsg.) Europäisches Vertragsrecht, Baden-Baden 1997
- Weyers, Hans-Leo Typendifferenzierung im Werkvertragsrecht, in: AcP
182 (1982), S. 60

Z

- Zehnder, Carl August Informatik-Projektentwicklung, 2. Aufl., Stuttgart 1991
- Zilahi-Szabó, Miklos G. Wirtschaftsinformatik, München Wien 1993
- Zweigert , Konrad/Kötz, Hein Einführung in die Rechtsvergleichung auf dem Gebiete
des Privatrechts, 3. Aufl., Tübingen 1996