

Brand, Andreas

**Book**

## Softwareentwicklung im Netzwerk: Kooperation, Hierarchie und Wettbewerb in einem Open Source-Projekt

**Provided in Cooperation with:**

Rainer Hampp Verlag

*Suggested Citation:* Brand, Andreas (2009) : Softwareentwicklung im Netzwerk: Kooperation, Hierarchie und Wettbewerb in einem Open Source-Projekt, ISBN 978-3-86618-327-8, Rainer Hampp Verlag, München

This Version is available at:

<https://hdl.handle.net/10419/68569>

**Standard-Nutzungsbedingungen:**

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

**Terms of use:**

*Documents in EconStor may be saved and copied for your personal and scholarly purposes.*

*You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.*

*If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.*

*Andreas Brand:*

## **Softwareentwicklung im Netzwerk. Kooperation, Hierarchie und Wettbewerb in einem Open Source-Projekt**

ISBN 978-3-86618-327-8, Rainer Hampp Verlag, München u. Mering, 2009, 303 S., € 29.80

In relativ kurzer Zeit haben Open Source-Projekte als selbst organisierte, informale (Arbeits-)Netzwerke freiwillig Arbeitender, die durch eine rechtlich spezifische Lizenz ein „öffentliches Gut“ erstellen, neben kommerzieller Software eine hohe Bedeutung erlangt. In dieser Arbeit wird die Zusammenarbeit der Open Source-Mitarbeiter bei der Softwareentwicklung anhand einer qualitativen Fallstudie des Open Source-Projekts KDE analysiert, das eine grafische Benutzeroberfläche besonders für Linux erstellt. Als zentrale Fragestellung wird die Koordination in den Arbeitsprozessen des Projekts untersucht. Daneben wird auf den Einfluss der Informations- und Kommunikationstechnik (IK) auf die hoch virtuelle Zusammenarbeit der Softwareentwicklung eingegangen. In der Literatur ging man bisher davon aus, dass die Organisationsform Netzwerk (ähnlich wie in Unternehmen und Märkten) nur durch eine Koordinationsform, in diesem Fall Kooperation, bestimmt wird. In dieser Studie kann nachgewiesen werden, dass neben Kooperation auch Hierarchie und Wettbewerb als Koordinationsformen in bestimmten Teilprozessen der Softwareentwicklung eine manchmal auch bedeutende Rolle spielen. Ausgehend von diesem Befund werden erste Anstrengungen durchgeführt, eine Theorie der Koordination zu erstellen. Zugleich werden Hypothesen und Vermutungen über die Ausprägung und die Bedeutung der Koordinationsformen im Open Source-Projekt KDE anhand der Neuen Institutionenökonomie und des soziologischen Neo-Institutionalismus aufgestellt.

**Schlüsselwörter:** Arbeit, Informations- und Kommunikationstechnologie, Netzwerk, Softwareentwicklung, Open Source, Koordination, Kooperation, Hierarchie, Wettbewerb, Organisationstheorie, Institutionentheorie, Neo-Institutionalismus, Neue Institutionenökonomie, Organisationsform, Informale Organisation, virtuelle Gruppe, qualitative Forschung

Andreas Brand hat 2007 an der Universität Frankfurt in Soziologie promoviert. Während seiner Promotion war er wissenschaftlicher Mitarbeiter in dem DFG-Projekt "Elektronische Arbeitsmärkte" (PELM) am Institut für Gesellschafts- und Politikanalyse, Schwerpunkt Wirtschaft, der Johann Wolfgang Goethe-Universität. Zurzeit ist er Referent in einem Industrieverband.

Thematische Schwerpunkte: qualitative und quantitative empirische Arbeitsmarktforschung, auch im Zusammenhang mit Informations- und Kommunikationstechnologie, Organisations- und Institutionentheorien, Neue Institutionenökonomie, Neo-Institutionalismus, Koordination und Koordinationsformen, Arten der Zusammenarbeit, Funktionsweise informeller sowie virtueller Gruppen wie Selbständigen-Netzwerken und Open Source-Projekten, Verbände, Methoden der Moderation von Kleingruppen.



Andreas Brand

# Softwareentwicklung im Netzwerk

Kooperation, Hierarchie und Wettbewerb  
in einem Open Source-Projekt

### **Bibliografische Information der Deutschen Nationalbibliothek**

Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.ddb.de> abrufbar.

ISBN 978-3-86618-327-8

DOI 10.1688/9783866183278

1. Auflage, 2009

Zugl.: Dissertation, Univ. Frankfurt, 2008, D.30

© 2009 Rainer Hampp Verlag München und Mering  
Marktplatz 5 D – 86415 Mering  
[www.Hampp-Verlag.de](http://www.Hampp-Verlag.de)

Alle Rechte vorbehalten. Dieses Werk einschließlich aller seiner Teile ist urheberrechtlich geschützt. Jede Verwertung außerhalb der engen Grenzen des Urheberrechtsgesetzes ist ohne schriftliche Zustimmung des Verlags unzulässig und strafbar. Das gilt insbesondere für Vervielfältigungen, Mikroverfilmungen, Übersetzungen und die Einspeicherung in elektronische Systeme.

∞ *Dieses Buch ist auf säurefreiem und chlorfrei gebleichtem Papier gedruckt.*

*Liebe Leserinnen und Leser!*

*Wir wollen Ihnen ein gutes Buch liefern. Wenn Sie aus irgendwelchen Gründen nicht zufrieden sind, wenden Sie sich bitte an uns.*

Für Nicole und Kilian



# Vorwort

Diese Arbeit ist aus dem Forschungsprojekt „Project Electronic Labour Markets“ (PELM) an der Johann-Wolfgang-Goethe-Universität zu Frankfurt am Main hervorgegangen. Den Weg zu einer abgabefertigen Arbeit haben zahlreiche Personen begleitet.

Viele Ergebnisse dieser Dissertation wären nicht ohne die vielen Anregungen und Hilfestellungen meines Doktorvaters, Prof. Dr. Alfons Schmid, und meines Zweitgutachters, Prof. Dr. Rudi Schmiede, entstanden, denen mein ganz besonderer Dank gilt.

Herzlich danken möchte ich allen Interviewpartnern und den Mitgliedern des Open Source-Projekts KDE, ohne die diese Arbeit nicht hätte entstehen können.

Meinen Kollegen Claus Becher und André Kawai aus dem Projekt PELM danke ich für die interessanten Diskussionen. Ein herzliches Dankeschön auch an alle Kolleginnen und Kollegen des IWAK-Instituts, Frankfurt und der Universität Frankfurt.

Für die methodische, theoretische und wissenschaftliche Unterstützung möchte ich mich bei Prof. Dr. Birgit Blättl-Mink und Prof. Dr. Günter Ropohl bedanken. Hinweise zur Informatiker-Profession habe ich von Andrea Sieber und Peter Bittner erhalten. Vielen Dank dafür.

Dr. Ulle Jäger, Dr. Tomke König und Dr. Rafael Behr sei für ihre Anleitung und Unterstützung gedankt. Weiterer Dank gebührt ferner folgenden Kolleginnen und Kollegen aus der Forschungssupervision und Textwerkstatt, aus der ich viele hilfreiche Anregungen mitgenommen habe: Barbara Hyna, Inga Halwachs, Tanja Becker, Andrea Neugebauer, Eran Gündüz und Hellen Fitsch.

Ferner möchte ich mich bei folgenden Freunden und Kollegen für deren Hilfe, den wissenschaftlichen Austausch und die sorgfältigen Korrekturarbeiten bedanken:

Dr. Kajetan Hinner, Dr. Gregor Leclerque, Jörg Rossow, Dr. Markus Grüne, PD Dr. Ursula Holtgrewe und Nils Beckmann.

Mein besonderer Dank für die tatkräftige Unterstützung gebührt schließlich meiner Partnerin Nicole, meinem Sohn Kilian, der uns so viel Freude bereitet, meinen Eltern und meinen Freunden Felix Schäfer, Susanne Dieckmann, Thomas Pühler, Anke Lundershausen und Michael Jarczyk. Zudem danke ich Tom Modos für den Cartoon.

Mainz, im April 2007

Andreas Brand

# Inhaltsverzeichnis

<b>1.</b>	<b>Einleitung.....</b>	<b>1</b>
1.1.	Problemaufriss .....	1
1.2.	Problemstellung und Zielsetzung.....	2
1.3.	Aufbau der Arbeit.....	7
 <b>ABSCHNITT I. ERSTELLUNG EINES UNTERSUCHUNGSRAHMENS: ARBEITSPROZESS UND KOORDINATION</b>		
<b>2</b>	<b>Virtuelle Arbeitsnetze als neue Arbeitsform: Arbeitsprozess der Softwareentwicklung und Rolle der Technik.....</b>	<b>9</b>
2.1	Virtuelle Arbeitsnetze als neue Arbeitsform .....	10
2.1.1	Virtuelle Arbeitsnetze - Zwischen Erwerbs-, Eigen-, freiwilliger und Ehrenamtsarbeit .....	10
2.1.2	Open Source-Projekte als virtuelle Arbeitsnetze .....	13
2.2	<b>IuK-Technik und Arbeitsprozess in virtuellen Arbeitsnetzen.....</b>	<b>16</b>
2.2.1	IuK-Technik als Gestell bzw. als Black Box.....	17
2.2.2	IuK-Technik als Medium.....	17
2.2.3	Zusammenhang zwischen IuK-technischer Infrastruktur und Softwareentwicklung	18
2.3	<b>Entwicklung eines idealisierten Arbeitsprozesses als Analyserahmen für die Softwareentwicklung.....</b>	<b>21</b>
2.3.1	Arbeitsmarktfunktionen und implizit angenommener Arbeitsprozess.....	22
2.3.2	Verschiedene Phasenmodelle der Softwareentwicklung .....	23
2.3.3	Teilprozesse des Softwareentwicklungsprozesses als Analyserahmen .....	30
2.4	<b>Zusammenfassung .....</b>	<b>33</b>
<b>3</b>	<b>Koordination und Arbeitsnetze als Organisationsform – Stand der Forschung und (Neu-)Definition .....</b>	<b>36</b>
3.1	<b>Diskurs über Koordination und Organisation aus der Netzwerkperspektive – Abgrenzung von Koordinationsform und Organisationsform .....</b>	<b>37</b>
3.1.1	Methodische, formale Netzwerkanalyse .....	38
3.1.2	Netzwerke im empirischen Diskurs .....	40
3.1.3	Idealtypisierte Netzwerke im theoretischen Diskurs - Stand der Forschung.....	42
3.1.4	Weitere Entwicklungen innerhalb Unternehmen, Märkten und Netzwerken.....	45
3.2	<b>Koordination und Organisation als Institutionen – Weiterführende Überlegungen und Neudefinition .....</b>	<b>48</b>
3.2.1	Definition des Institutionenbegriffs.....	49
3.2.2	Definition der Organisationsformen .....	50
3.2.3	Definition der Koordinationsformen .....	53
3.3	<b>Zusammenfassung .....</b>	<b>73</b>

## **ABSCHNITT II. KOORDINATIONSFORMEN IM EMPIRISCHEN FELD**

<b>4</b>	<b>Konzept und Methodik der empirischen Untersuchung .....</b>	<b>77</b>
<b>4.1</b>	<b>Methodisches Vorgehen.....</b>	<b>77</b>
4.1.1	Auswahl der Untersuchungsobjekte .....	77
4.1.2	Datenerhebung .....	78
4.1.3	Datenaufbereitung.....	79
<b>4.2</b>	<b>Bewertung der empirischen Daten: Indikatoren für Koordinationsformen.....</b>	<b>79</b>
4.2.1	Institutionalisierungsgrad als Indikator.....	80
4.2.2	Institutionalisierungsgrad der Koordinationsformen .....	80
4.2.3	Messung des Institutionalisierungsgrads in den Teilprozessen der Softwareentwicklung .....	82
<b>5</b>	<b>Koordinierung des Arbeitsprozesses im Open Source-Projekt KDE.....</b>	<b>84</b>
<b>5.1</b>	<b>Annäherung an Open Source-Projekte über die Open Source-Bewegung .....</b>	<b>84</b>
5.1.1	Open Source-Bewegung als soziale Bewegung.....	85
5.1.2	Akteure der Open Source-Bewegung, darunter Open Source-Projekte.....	88
<b>5.2</b>	<b>Akteure des Open Source-Projekts KDE .....</b>	<b>94</b>
5.2.1	Sozioökonomische Charakteristika .....	94
5.2.2	Motivation .....	96
5.2.3	Tätigkeits- und Strukturbereiche.....	98
<b>5.3</b>	<b>IuK-technische Infrastruktur des Open Source-Projekts KDE.....</b>	<b>99</b>
5.3.1	Kommunikationsinfrastruktur.....	102
5.3.2	Informationsinfrastruktur.....	103
5.3.3	Produktionsinfrastruktur .....	103
<b>5.4</b>	<b>Koordinationsformen im Open Source-Projekt KDE .....</b>	<b>104</b>
5.4.1	Koordination in Open Source-Projekten – Stand der Forschung .....	104
5.4.2	Zusammenhang zwischen IuK-technischer Infrastruktur und Koordinationsformen - Stand der Forschung.....	107
5.4.3	Koordinationsformen in den Teilprozessen der Softwareentwicklung .....	108
<b>5.5</b>	<b>Zusammenfassung .....</b>	<b>126</b>

### **ABSCHNITT III. ANALYTIK DER VORGEFUNDENEN KOORDINATIONSFORMEN IM OPEN SOURCE-PROJEKT KDE**

<b>6. Analytik der Koordinationsformen im Open Source-Projekt KDE anhand von Institutionentheorien .....</b>	<b>132</b>
<b>6.1. Analyse der Koordinationsformen anhand der Neuen Institutionenökonomie.....</b>	<b>135</b>
6.1.1. Theorieinterpretation der Neuen Institutionenökonomie - Theoretische Grundlagen der Analyse .....	138
6.1.2. Analyse der Koordinationsformen im Open Source-Projekt KDE.....	163
6.1.3. Zusammenfassung: Analyse der Koordinationsformen anhand der Neuen Institutionenökonomie .....	183
<b>6.2. Analyse der Koordinationsformen anhand des Neo-Institutionalismus .....</b>	<b>186</b>
6.2.1. Theorieinterpretation des Neo-Institutionalismus - Theoretische Grundlagen der Analyse .....	190
6.2.2. Analyse der Koordinationsformen im Open Source-Projekt KDE.....	207
6.2.3. Zusammenfassung: Analyse der Koordinationsformen anhand des Neo-Institutionalismus .....	244
<b>7. Zusammenfassung und Ausblick .....</b>	<b>248</b>
<b>7.1. Koordination der Softwareentwicklung in einem Open Source-Projekt .....</b>	<b>248</b>
<b>7.2. Zusammenhang zwischen IuK-Technik und Koordinationsformen sowie zwischen Endnutzern und Softwareentwicklern .....</b>	<b>252</b>
<b>7.3 Analytik der vorgefundenen Koordinationsformen im Open Source-Projekt KDE.....</b>	<b>254</b>
<b>8. Literatur .....</b>	<b>258</b>
<b>9. Anhang.....</b>	<b>278</b>
<b>Anhang A: Tabellen und Abbildungen.....</b>	<b>278</b>
<b>Anhang B: Interviewleitfaden für das Open-Source-Projekt.....</b>	<b>281</b>
<b>Anhang C: Quantitativer Fragebogen.....</b>	<b>286</b>
<b>Anhang D: Kategorien der Auswertung der Selbstaussagen des KDE-Projekts.....</b>	<b>291</b>

## Abbildungsverzeichnis

Abb. 2-1:	Phasenabfolge des Wasserfallmodells .....	25
Abb. 2-2:	Entwicklungsphasen in einem Spiralmodell .....	26
Abb. 2-3:	Ablauf eines Open Source-Projekts .....	28
Abb. 2-4:	Analyserahmen des Arbeitsprozesses der Softwareentwicklung mit Teilprozessen .....	31
Abb. 3-1:	Organisationsformen der Koordination.....	57
Abb. 6-1:	Entstehung initialer Transaktionsprobleme, eigene Darstellung .....	141
Abb. 6-2:	Initiale Transaktionsprobleme und Beherrschungs/ Überwachungssysteme .....	148
Abb. 6-3:	Unterschiedlicher Einfluss von Kooperation, Hierarchie/Macht und Wettbewerb als Koordinationsformen auf die Transaktionskosten .....	152
Abb. 6-4:	Parsons Gesellschaftssystem mit seinen vier Hauptsystemen inkl. Funktionen.....	204
Abb. 6-5:	Modell des Einflusses der Strukturangleichungsmechanismen und der Ressourcen.....	205
Abb. 6-6:	Verortung des Open Source-Projekts KDE und der beeinflussenden gesellschaftlichen Umweltbereiche in Parsons Gesellschaftssystem .....	234

### Anhang A

Abb. A-1:	Verteilung der Mitarbeit bei KDE, Mitte 2003, 740 Entwickler .....	280
Abb. A-2:	Häufigkeit der Nennung der Mitarbeiteranzahl an Subprojekten bei KDE (quant. Befragung), absolut .....	280

## Tabellenverzeichnis

Tab. 2-1:	Anpassungsschritte zum Erhalt eines idealisierten Arbeitsprozesses mit Teilprozessen .....	30
Tab. 3-1:	Typologie von Koordinationsformen.....	44
Tab. 3-2:	Elemente der Koordinationsformen anhand des Kategorisierungsschemas .....	75
Tab. 5-1:	Bedeutung der Koordinationsformen in den Teilprozessen der Softwareentwicklung des Open Source-Projekts KDE .....	127
Tab. 6-1:	Effizienz der jeweiligen Koordinationsform durch Einfluss auf die jeweiligen Transaktionskosten.....	172

## Anhang A

Tab. A-1:	Anzahl der Befragten nach Nationalität der Kommunikationspartner und der Personen der Selbstaussagen nach eigener Nationalität, absolut .....	278
Tab. A-2:	Anzahl der Befragten und der Personen der Selbstaussagen nach IT-Ausbildung/Studium, absolut.....	278
Tab. A 3:	Anzahl der Befragten und der Personen der Selbstaussagen nach Geschlecht, absolut.....	279
Tab. A 4:	Anzahl der Befragten und der Personen der Selbstaussagen nach Geburtsjahrkategorie, 2003, absolut .....	279
Tab. A 5:	Anzahl der Befragten und der Personen der Selbstaussagen nach Tätigkeiten, absolut .....	279

## Abkürzungsverzeichnis

CSCW	Computer Supported Cooperative Work
F.u.E.	Forschung und Entwicklung
FSF	Free Software Foundation
GPL	General Public License
IuK-Technik	vernetzte und vernetzende Informations- und Kommunikationstechnik
KDE	K Desktop Environment (Name des Open Source-Projekts und seines Produkts)
LGPL	Lesser General Public License
MOO	MUD object oriented
MUD	Multi-User-Dungeon

# 1. Einleitung

## 1.1. Problemaufriss

Betrachtet man die gesellschaftliche Arbeit, so fällt einerseits eine Differenzierung der Beschäftigungsverhältnisse (vgl. Willke 1998), andererseits auch eine Diversifizierung ihrer Form auf. Nicht nur selbständige Arbeitsformen (vgl. z.B. Leicht, Philipp 1999) oder Telearbeit (vgl. z.B. Baukrowitz et al. 1998) auf Basis von vernetzter und vernetzender Informations- und Kommunikationstechnik (IuK-Technik) werden bedeutender, als auch Nicht-Erwerbsarbeit wird bedeutender und die Grenze zwischen Ehrenamts-, Eigen- und Erwerbsarbeit verändert sich (vgl. z.B. Voß, Rieder 2005). Im Zusammenhang mit IuK-Technik und ggf. der erhöhten Selbständigkeit schließen sich Selbständige, aber auch Freiwillige, vermehrt zu Arbeitsnetzen zusammen. Gerade in der Softwareentwicklung ist dies der Fall (vgl. Henninger, Sieber 2001), tritt aber auch in der Film- und Fernsehbranche auf (vgl. Blair 2003). In Arbeitsnetzen arbeiten autonome, überwiegend individuelle Akteure temporär und informal zusammen und erstellen ein gemeinsames Produkt (vgl. Brand 2004).

Seit den 1990er Jahren hat die Vernetzung zwischen den Computern über IuK-Technik zugenommen, was zu einer weiteren Veränderung der gesellschaftlichen Arbeit führte, aber vor allem neue Arbeitsformen wie virtuelle Arbeitsnetze aufkommen ließ. Gerade die virtuellen Arbeitsnetze sind interessant, da sie sich stark von der Erwerbsarbeit in Unternehmen unterscheiden und bisher von der wissenschaftlichen Forschung eher vernachlässigt wurden.

„Insgesamt handelt es sich bei diesen Zusammenhängen zwischen praktischer Kooperation, Netzwerkformen, Nutzung von IuK-Techniken, Wissenstransfer und Arbeit jedoch um ein wenig untersuchtes Gebiet, d.h. es besteht ein ausgeprägtes Forschungsdefizit. Will man einen empirisch reichen qualitativen Begriff der informatisierten Arbeit entwickeln, wird man sich auf die mühsame Erkundung dieser Detailebene von Arbeit und Kooperation einlassen müssen“ (Schmiede 2006: 468).

Auf dieser Detailebene von Arbeit und Kooperation werden in der vorliegenden Arbeit erste Schritte unternommen, anhand derer weitere Untersuchungen zum qualitativen Begriff informatisierter Arbeit nachfolgen können. Unter dem Begriff informatisierte Arbeit wird „die Prozesshaftigkeit der Durchdringung aller gesellschaftlichen Dimensionen mit neuen Inhalten, Formen und Techniken der Information“ verstanden (Schmiede 2006: 458).

Bei Betrachtung der IuK-technischen Virtualisierung der Arbeit als Teilbereich der informatisierten Arbeit wird überwiegend die Veränderung der Arbeitsweise bei der Erwerbsarbeit in Unternehmen z.B. in standortverteilten Gruppen oder der Bezug zwischen Arbeitswelt und Lebenswelt untersucht (vgl. z.B. Huber, Hirschfelder 2004; Baukrowitz et al. 2006; Goll 2002; Boos et al. 2000). Hervorstechend sind die angesprochenen Arbeitsnetze, die eine Software entwickeln und hoch virtuell sind. Sie kommunizieren und produzieren zeitlich und örtlich ungebunden überwiegend mittels IuK-Technik ein gemeinsames Softwareprodukt. Diese hohe Virtualität in Kombination mit einer neuen temporär-informalen Arbeitsform rückt den Arbeitsprozess außerhalb der Projektarbeit in Unternehmen in den Blickpunkt des Interesses, da bisher in Studien über Softwareentwicklung fast nur Unternehmen sowie Phasenmodelle, Arbeitsstile und Projekttypen im Mittelpunkt standen (vgl. z.B. Welz, Ortmann 1992; Konrad, Gerd 1999; Bittner et al. 1995; Henninger, Sieber 2001).

Beispiel für diese neue, vollständig virtuelle Arbeitsform Arbeitsnetz sind Open Source-Projekte<sup>1</sup>, bei denen eine Softwareentwicklung stattfindet. In Open Source-Projekten wird freiwillig im Gegensatz zur virtuellen Erwerbsarbeit zusammengearbeitet, was diese Projekte besonders interessant macht. Sie werden daher in dieser Arbeit untersucht. Die Open Source-Projekte sind Ausdruck eines allgemeinen Trends, der sich im Internet in Form von virtuellen Gruppen darstellt. Diese virtuellen Gruppen eint neben dem Wissensaustausch eine gemeinsame Produkterstellung als gemeinsames Interesse, unterstützt durch Foren, Usenet oder Mailinglisten (vgl. Götzenburger, Löger 2000; Döring 2003: 489ff.).

Zusätzlich werden in einem Open Source-Projekt Softwareprodukte erstellt, die den Produkten der kommerziellen Softwareunternehmen in nichts nachstehen. Während kommerzielle Softwareunternehmen Lizenzgebühren für ihre proprietären Softwareprodukte fordern, stehen die Open Source-Softwareprodukte durch eine rechtlich-innovative Lizenz als „öffentliches Gut“ allen Nutzern offen. Open Source-Projekte sind Teil einer sozialen Bewegung, in der beispielsweise das bekannte, untersuchenswerte KDE-Projekt eine grafische Benutzeroberfläche vor allem für das Open Source-Betriebssystem Linux produziert. Damit steht die freiwillige Arbeit in einem Open Source-Projekt im Zusammenhang mit Eigen- und Ehrenamtsarbeit ohne Entlohnung. Durch die Freiwilligkeit der Arbeit unterscheidet sich die Arbeitsweise in den meisten Open Source-Projekten stark von der kommerziellen Softwareentwicklung in Unternehmen. Dabei hat diese innovative Art der Softwareerstellung ab den 1990er Jahren eine starke Verbreitung erfahren (vgl. Holtgrewe, Werle 2001) und kann in ihrem Erfolg mit der Softwareentwicklung in Unternehmen verglichen werden. Die Genese der Open Source-Bewegung und –Projekte wird jedoch in dieser Untersuchung nicht betrachtet.

## 1.2. Problemstellung und Zielsetzung

Beim Open Source-Projekt als virtuelles Arbeitsnetz interessiert der Arbeitsprozess der Softwareentwicklung. Bisher wurde die Softwareentwicklung, wie schon erwähnt, auf einer individuellen Ebene in Form von Arbeitsstilen sowie auf einer umfassenden (Projekt-)Gesamtebene der Softwareentwicklung in Form von Phasenmodellen und Projekttypen betrachtet (vgl. z.B. Welz, Ortmann 1992; Konrad, Gerd 1999; Bittner et al. 1995; Henninger, Sieber 2001). Phasenmodelle sind dabei Vorgehensmodelle bzw. Vorstellungen eines idealen zeitlichen Arbeitsablaufs in der Softwareentwicklung, die weitestgehend der Komplexität der realen Softwareentwicklung im empirischen Feld z.B. im Open Source-Projekt nicht vollständig gerecht werden. Sie bilden eher bestimmte Kernziele eines Arbeitsprozesses ab. Besonders in der Literatur über Open Source-Projekte, die im Mittelpunkt dieser Untersuchung stehen, werden die Motivationen der Softwareentwickler auf einer individuellen Ebene sowie Softwareentwicklungsphasen oder Normen und Werte wie Hackerethik, technische Rationalität, Führungsstruktur und Reputation auf einer übergreifenden Projektebene festgestellt (vgl. Weber, S. 2004: 157ff.; Holtgrewe, Brand 2007). Eine Perspektive, die Handlungen zwischen diesen Ebenen, z.B. Arbeitsprozesse, beleuchtet,

---

<sup>1</sup> Üblicherweise wird dieses Phänomen Free Software oder Open Source, von europäischen Wissenschaftlern auch Libre Software, als eher neutraler, aber inhaltsleerer Begriff genannt. Die Bezeichnungen beziehen sich auf die freie Verfügungsmöglichkeit über Software. Es wird hier die Verwendung von Open Source als Kennzeichnung aller Free Software-/Open Source-Projekte benutzt, da sich dieser Begriff weitestgehend durchgesetzt hat. Der gängige Open Source-Begriff wurde auch gewählt, da er die Tatsache des technisch frei zugänglichen Quellcodes beschreibt und den meisten Projekten zu eigen ist. Bei genauer Betrachtung der Open Source-Bewegung kann eine pragmatische, produktorientierte Strömung mit dem Begriff „Open Source“ und eine normative Strömung mit „Free Software“ unterschieden werden. Mit der Verwendung dieser Bezeichnung ist jedoch keine Parteilichkeit für eine der Vorstellungen verbunden.

wurde bisher in der Literatur ebenso wenig verfolgt wie die Frage, wie und in welchen Situationen Normen bzw. Institutionen während des Arbeitsprozesses der Softwareentwicklung Anwendung finden. Sie stehen aus diesem Grund im Mittelpunkt dieser Arbeit. Da Normen in spezifischen Situationen auftreten, werden als Teile des Arbeitsprozesses logisch-abfolgende Teilprozesse der Softwareentwicklung als situative Elemente entwickelt. In den Open Source-Projekten wird eher parallel und nicht in aufeinanderfolgenden Phasen gearbeitet (vgl. Jörgensen 2001; Mockus et al. 2002). Deshalb werden verschiedene Ansätze zur Beschreibung der Teilprozesse herangezogen und ein neuer Ansatz verfolgt. Der neue analytische Ansatz der logisch-abfolgenden Teilprozesse ist interessant, da bisher nur zeitlich abfolgende Phasen in Vorgehensmodellen betrachtet wurden.

Nicht nur Open Source-Projekte als Arbeitsnetze, sondern auch andere Netzwerke sind seit den 1990er Jahren verstärkt in den wissenschaftlichen Blickpunkt gerückt. Netzwerke zwischen Unternehmen, aber auch mit staatlichen Akteuren breiteten sich aus. Ein Grund für ihre Ausbreitung sind die sich häufig ändernden, krisenhaften sozio-ökonomischen Entwicklungstendenzen, die durch eine Verbreitung der vernetzten und vernetzenden IuK-Technik, einer neuen Unmittelbarkeit des Marktprinzips in einer globalisierten Welt und einer Krise der bürokratischen hierarchischen Steuerung im Unternehmen entstehen. Ein weiterer Grund sind Konflikte zwischen gesellschaftlichen Teilsystemen und stringent verfolgten Handlungslogiken der Unternehmen (Sauer 2001: 30ff.; Hirsch-Kreinsen 2002: 106ff. Schimank 2005: 39ff.). Mit der Verbreitung der Netzwerke ging auch eine vermehrte Auseinandersetzung mit diesen in der Literatur einher. Da Arbeitsnetze eine Unterform der Netzwerke sind, ist der wissenschaftliche Netzwerkdiskurs genauer zu betrachten.

In der wissenschaftlichen Diskussion werden Netzwerke einerseits weitgehend als Organisationsform zwischen bzw. jenseits von den Organisationsformen Markt und Unternehmen betrachtet. Sie bestehen in einer vorgreifenden Definition aus autonomen Akteuren, die selbstorganisiert, wiederholt temporär gemeinsam Vorteile über informale Beziehungen realisieren, aber einem begrenzten, exklusiven Zugang unterliegen. Arbeitsnetze sind Netzwerke, in denen als vorteilhaftes Ziel ein gemeinsames Produkt durch gemeinsames Arbeiten angestrebt wird (s.o.).

Andererseits wird die Koordination von und in Netzwerken diskutiert (vgl. Wiesenthal 2005; Müller-Jentsch 2003: 131ff.; Sydow, Windeler 2000). Koordination ist in einer vorgreifenden Definition ein wechselseitiges aufeinander Abstimmen bzw. Synchronisieren von Interaktionen und Erwartungen, die zu gerichteten Handlungen führen. Sie ist eine Institution, die aus einem Bündel von Normen und Regeln besteht. Die Normen des Arbeitsprozesses der Softwareentwicklung liegen auf einer individuellen Handlungsebene und stehen damit im Fokus der Untersuchung. Daher wird untersucht, wie der Arbeitsprozess der Softwareentwicklung mit seinen Teilprozessen koordiniert wird. Die Untersuchung der Koordination in einem spezifischen Netzwerk wird damit auch einen Beitrag zur Netzwerkdiskussion liefern. Es wird ein Open Source-Projekt herausgegriffen, an dem die Koordination explorativ untersucht werden kann. Weitere Netzwerke werden in dieser Arbeit nicht untersucht.

Koordinationsformen sind spezifische Institutionen der Koordination und ein zentraler Begriff für diese Arbeit. In dem Netzwerkdiskurs, der auch Markt und Unternehmen behandelt, wird nicht zwischen Organisations- und Koordinationsform unterschieden. Organisationsformen werden als Koordinationsformen aufgefasst (vgl. z.B. Wiesenthal 2005; Kawai, Schmid 2004). Die Organisationsform Netzwerk wird mit Kooperation als Koordinationsform, das Unternehmen mit Hierarchie und der Markt mit Wettbewerb gleichgesetzt. Für eine Untersuchung des Arbeitsprozesses der Softwareentwicklung sind aber die Handlungen auf der individuellen, nicht der organisationellen oder gesellschaftlichen Ebene rele-

vant. Zusätzlich wurden im empirischen Feld verschiedene Koordinationsformen bzw. eine Koordinationsformenkombination in den Organisationsformen festgestellt, z.B. sind in Unternehmen neben hierarchischen Elementen kooperative Elemente wie Projektarbeit und wettbewerbliche Elemente wie leistungsorientierte Entlohnung anzutreffen (vgl. Picot et al. 1996; Zenger 2002; Moldaschl, Sauer 2000; Kawai, Schmid 2004; Kühl 1999; Hirsch-Kreinsen 1995; Wiesenthal 2005). Auch für atypische Märkte wie den elektronischen Arbeitsmarkt oder den „Gesundheitsmarkt“ wurden Kombinationen von Koordinationsformen festgestellt (vgl. Becher et al. 2007; Sachverständigenrat Gesundheitswesen 2005).

Vor diesem Hintergrund sind die drei Koordinationsformen „*Kooperation*“, „*Hierarchie*“ und „*Wettbewerb*“ von den Organisationsformen zu trennen und werden als auf einer Ebene liegend betrachtet. In der vorliegenden Arbeit ist also zwischen der Organisation als Institution und den institutionalisierten Regelungen *innerhalb* einer Organisation zu unterscheiden. Damit wird eine allgemeine lexikalische organisationstheoretische Definition verwendet (vgl. Fuchs-Heinritz et al. 1995: 302, 478; Hillmann 1994: 374f., 638f.), wobei Organisationen als künstlich hergestellte, dauerhafte zielgerichtete Institutionen mit Zugehörigkeitsgrenze aufgefasst werden. Institutionen in Organisationen dagegen sind formale und informale Regelungen, die das Handeln der Akteure in der Organisation direkt beeinflussen. Die Koordinationsformen heißen hier „*Kooperation*“, „*Hierarchie/Macht*<sup>2</sup>“ und „*Wettbewerb*“ und werden als institutionalisierte Regelungen innerhalb einer Organisation betrachtet. Der Fokus liegt also nicht auf der Organisationsform, sondern darauf, wie in einer Organisationsform mittels Kooperation, Hierarchie oder Wettbewerb der Arbeitsprozess koordiniert wird. Von besonderem Interesse ist dabei, wie die Koordination in den hoch virtuellen Arbeitsnetzen mit Softwareentwicklung erfolgt. Sie bildet in der vorliegenden Arbeit einen besonderen Schwerpunkt.

Mit dieser bisher unüblichen Perspektive wird der unübersichtlichen Darstellung der Koordination, z.B. bei den Unternehmens-, Innovationsnetzwerken etc., entgegengetreten (s. Müller-Jentsch 2003: 131ff.). So wird nicht der in der Ökonomie gebräuchlichen Sichtweise gefolgt, bei der die Koordinationsmechanismen Hierarchie und Wettbewerb eine Kooperation herbeiführen (vgl. Coase 1937; Williamson 1985), sondern ein interdisziplinärer Blick auf die Koordination geworfen.

Die zentrale Fragestellung dieser Arbeit ist somit, *wie die Teilprozesse der Softwareentwicklung in einem virtuellen Arbeitsnetz mittels Kooperation, Hierarchie/Macht und Wettbewerb koordiniert werden*. Auf welche Weise die Koordination in einem Open Source-Projekt erfolgt, kann anhand der Ausprägungen und des Stellenwerts der Koordinationsformen beantwortet werden. Damit wird die zentrale Fragestellung nach den *Ausprägungen und dem Stellenwert* von Kooperation, Hierarchie/Macht und Wettbewerb im Arbeitsprozess der Softwareentwicklung mit seinen Teilprozessen eines virtuellen Arbeitsnetzes präzisiert. Unter Ausprägung der Koordinationsform werden die Elemente der Koordinationsformen verstanden. Der Stellenwert bzw. die Bedeutung einer Koordinationsform wird durch ihre Dominanz bzw. ihr deutlich häufigeres Auftreten gegenüber den anderen Koordinationsformen festgestellt. Dabei wird ein statischer, idealisierter Blick eingenommen. Die historische Genese und die zeitliche Veränderung der Koordinationsformen werden nicht betrachtet.

---

<sup>2</sup> Hierarchie bzw. Macht tritt in einer statischen und einer dynamischen Form auf, wobei Hierarchie eher mit der statischen und Macht mit der dynamischen Form verbunden wird. Beide Formen bezeichnen aber eine Koordinationsform, die „Hierarchie/Macht“ genannt wird, um dieser Doppelbedeutung gerecht zu werden.

Es wird also ein Open Source-Projekt als virtuelles Arbeitsnetz empirisch untersucht und Aussagen zur sozialen Koordination des Arbeitsprozesses getroffen. Über den Fokus auf die Softwareentwicklung und die Nebenfrage, wie sich Ausprägungen und Stellenwert der Koordinationsformen in diesem virtuellen Arbeitsnetz bestehen, kann ein Beitrag zum Diskurs über neue virtuelle Arbeitsformen und zu Netzwerken geleistet werden. Weitere Arbeitsabläufe oder Arbeitsformen, die nicht direkt mit der Softwareentwicklung verbunden sind, werden weitestgehend ausgeblendet. Es wird weder ein Vergleich zwischen virtuellen und nicht-virtuellen Arbeitsnetzen bzw. Arbeitsformen noch mit der Organisationsform Markt oder Unternehmen angestrebt.

Das untersuchte Arbeitsnetz ist durch eine hohe Virtualität gekennzeichnet, was Interesse für eine weitere Betrachtung der IuK-Technik weckt. Die hohe Virtualität hängt mit einer ständigen und umfassenden Verwendung der IuK-Technik zusammen, welche die Arbeitsnetze erst ermöglicht und damit auf die Rolle der IuK-Technik aufmerksam macht. Dabei drängt sich eine weitere Nebenfrage danach auf, wie diese IuK-Technik den Arbeitsprozess der Softwareentwicklung beeinflusst. Die zu den Arbeitsnetzen gehörende IuK-Technik setzt sich aus Teilen der Hard- und Software des Internets und der Arbeitsnetz-Beteiligten zusammen. Es wird einerseits Software erstellt, andererseits fungiert Software als Infrastruktur. Diese Zwitterfunktion der Software und die Neuheit dieser Technik mit ihren großen Auswirkungen auf soziale Beziehungen, aber auch Arbeitsabläufe (vgl. Döring 2003: 489ff.), führt zu einer Betrachtung der IuK-Technik aus einer technikphilosophischen und –soziologischen Perspektive. Die IuK-technische Infrastruktur in Form von Websites mit Foren, Mailinglisten, einem Versionsmanagementsystem und einem Chatsystem ist dabei als ein statisches, gekapseltes Artefakt zu sehen, das die Erstellung eines Softwareprodukts unterstützt. Obige Nebenfrage nach dem Einfluss der IuK-Technik auf die Softwareentwicklung wird umformuliert und auf die Koordination und den Arbeitsprozess bezogen. Es wird also einerseits untersucht, wie die IuK-Technik als statische, gekapselte Infrastruktur mit den Koordinationsformen Kooperation, Hierarchie/Macht und Wettbewerb der Softwareentwicklung im Open Source-Projekt und virtuellen Selbständigen-Arbeitsnetz zusammenhängt. Andererseits interessiert, welcher Zusammenhang mit den Teilprozessen der Softwareentwicklung besteht, was jedoch nur am Rande von Interesse ist. Ein Vergleich mit der Nutzung der IuK-technischen Infrastruktur im Unternehmen oder durch Privatpersonen im Internet erfolgt nicht, da diese spezifische Fragestellung keinen direkten Bezug zum Schwerpunkt dieser Arbeit hat.

Beim Fokus auf die Technikerstellung als Arbeitsprozess spielen aber nicht nur die Softwareentwickler als „Konstrukteure“, sondern auch die Endnutzer des Endprodukts eine Rolle. Diesmal stehen die Akteure der Softwareentwicklung im Arbeitsprozess im Zentrum. Die Softwareentwicklung ist in soziale Beziehungen eingebettet, in denen Interaktionen zwischen den Entwicklern, aber auch zwischen Entwicklern und Endnutzern bestehen (vgl. Glock 1997; Pinch, Bijker 1987). Daher wird auch der Frage nachgegangen, wie sich die Beziehung zwischen Endnutzern und den Softwareentwicklern darstellt. In der Beziehung kann beispielsweise Kooperation, Hierarchie/Macht oder Wettbewerb dominieren, aber auch eine andere Koordinationsformenkombination ist denkbar.

Die vorigen Fragestellungen werden in dieser Arbeit durch eine empirische, qualitative, explorative Fallstudie des Open Source-Projekts KDE beantwortet. Diese sind bisher noch nicht in dieser Form untersucht worden. Zentral waren leitfadengestützte Experteninterviews, die durch eine Dokumentenanalyse der Website und von Mailinglisten, teilnehmenden Beobachtungen auf Konferenzen etc. ergänzt wurden. Die Daten wurden im Rahmen des von der Deutschen Forschungsgemeinschaft geförderten Forschungsprojekts „Struktur und Funktionsweise elektronischer Arbeitsmärkte“ erhoben. Aus diesem Forschungsprojekt wurden verschiedene Artikel bzw. Bücher veröffentlicht, die das gleiche empirische

Material weitestgehend unter anderen Fragestellungen verwendeten (s. Becher et al. 2005; Brand, Schmid 2006; Brand, Holtgrewe 2007; Holtgrewe, Brand 2007).

Ausgehend von den empirischen Ergebnissen des Open Source-Projekts KDE als Untersuchungsgegenstand ergibt sich die Frage, warum die Koordinationsformen in den Teilprozessen der Softwareentwicklung die vorgefundenen Ausprägungen und Bedeutungen haben. Die genannte Frage wird in dieser Arbeit nicht durch deduktiv-nomologische Erklärungen beantwortet. Solche Erklärungen sind in der vorliegenden Arbeit nicht möglich, da bisher keine dominierende und einheitliche Theorie der Koordination in Organisationen besteht und aufgrund der Begrifflichkeiten und fehlender Begründungszusammenhänge vor allem im Bezug auf Institutionen auch nicht aus dem Netzwerkdiskurs erarbeitet werden kann. Statt dessen werden Vermutungen theoretisch hypothetischer Art anhand von anderen adäquaten Theorien erarbeitet, aus denen Hypothesen entwickelt werden können. Zentral ist dabei nicht der Zusammenhang zwischen Koordinationsformen und IuK-Technik oder die Beziehung zwischen Softwareentwicklern und Endnutzern, sondern das Auftreten von Kooperation, Hierarchie/Macht und Wettbewerb in den Teilprozessen der Softwareentwicklung.

Die gesuchten adäquaten Theorien sollen Institutionen und Organisationen umfassen und Koordinationsformen als Institutionen in ihre Theoriengefüge aufnehmen können. Die Neue Institutionenökonomie und der Neo-Institutionalismus wurden als Theorien ausgewählt, da sie sich als eine ökonomische bzw. eine soziologische Theorie mit Homo Oeconomicus bzw. Homo Sociologicus als Akteursvorstellungen ergänzen und zugleich auf Institutionen auf der Handlungs- und Mesoebene beziehen. Sie ergänzen sich auch, da die soziologische Theorie die Herkunft der Koordinationsformen benennen kann, die ökonomische aber nicht. Die Neue Institutionenökonomie bietet sich auch als eine Institutionentheorie an, da aus dieser Theorie die Diskurse um Organisationsformen wie Markt, Unternehmen und Unternehmensnetzwerke als Koordinationsformen entstanden. Die Neue Institutionenökonomie wurde auch erfolgreich zur Analyse von Koordinationsformen im elektronischen Arbeitsmarkt von Becher et al. (2007) verwendet. An erste Erkenntnisse zu einer Theoriemodifikation und eine Annäherung an eine Theorie der Koordination in Organisationen konnte dort angeknüpft werden. Die Erklärungskraft der Neuen Institutionenökonomie für kooperative Elemente wie Altruismus oder Vertrauen ist jedoch begrenzt, da diese über den Homo Oeconomicus nur kalkulierbares Vertrauen bzw. Kooperation betrachtet, und nicht-rationales prosoziales Handeln ausblendet. Des Weiteren wurden in der Neuen Institutionenökonomie hauptsächlich For-Profit-Unternehmen untersucht, aber eine Anwendung auf Non-Profit-Organisationen erfolgte mitunter (vgl. Theuvsen 1999). Als weitere Theorie wird der soziologische Neo-Institutionalismus herangezogen, der wissenschaftshistorisch als eine Gegenbewegung zur Neuen Institutionenökonomie entstand. Während in der Neuen Institutionenökonomie über den Aufwand (Transaktionskosten) argumentiert wird, steht beim Neo-Institutionalismus der gesellschaftliche Einfluss im Mittelpunkt. Die Institutionen in einer Organisation werden beim Neo-Institutionalismus aus der Einbettung dieser Organisation in die Gesellschaft bestimmt. Es wurden im Neo-Institutionalismus überwiegend Non-Profit-Organisationen untersucht, die Theorie wurde aber auch auf For-Profit-Unternehmen angewendet und dabei alternative Argumentationen entwickelt (vgl. Hasse, Krücken 1999: 7ff.; Fligstein 1985, 1991). Die drei Koordinationsformen können als Institutionen in diese Theorie übernommen werden, da im gesellschaftlichen Werte- und Normensystem kooperative, wettbewerbliche und hierarchische Elemente vorkommen (s. z.B. Thompson et al. 1990) und unterschiedlich Eingang in das Werte- und Normensystem von Organisationen finden können.

Beide Institutionentheorien werden derart modifiziert, dass alle drei Koordinationsformen im untersuchten Open Source-Projekt analysiert werden können. Anhand dieser modifi-

zierten Theorien können dann Vermutungen herausgearbeitet werden, die Erklärungsansätze für die vorgefundenen Ausprägungen und Bedeutungen der Koordinationsformen in den Teilprozessen der Softwareentwicklung liefern können. Zugleich können diese Theoriemodifikationen als eine Annäherung bzw. ein Schritt in Richtung einer zukünftigen einheitlichen Theorie der Koordination in Organisationen betrachtet werden. Es wird in dieser Arbeit zwar keine vollständige Theorie der Koordination in Organisationen angestrebt, allerdings wird analysiert, ob die modifizierten Theorien der Neuen Institutionenökonomie und des Neoinstitutionalismus für eine solche Theorie der Koordination in Organisationen verwendet werden können. Ein Effizienz-Vergleich der Softwareentwicklung zwischen virtuellen Arbeitsnetzen einerseits und zwischen virtuellen Arbeitsnetzen und anderen Organisationsformen wie Unternehmen andererseits erfolgt dabei ebenso wenig wie die Untersuchung des gesellschaftlichen Einflusses auf diese.

### 1.3. Aufbau der Arbeit

Die vorliegende Arbeit ist in drei Hauptabschnitte gegliedert, denen die Kapitel zugeordnet werden. Im *ersten Abschnitt* wird der Untersuchungsrahmen bzgl. des Arbeitsprozesses der Softwareentwicklung und Koordination erstellt, mit dem die späteren empirischen Untersuchungen erfolgen können. Zunächst wird in Kapitel 2 begründet, warum das Arbeitsnetz Open Source-Projekt ein lohnenswertes Untersuchungsobjekt darstellt. Danach folgt die theoretisch geleitete Reformulierung der obigen Frage nach dem Einfluss der IuK-Technik auf die Softwareentwicklung in dem Arbeitsnetz zur Frage nach der Beziehung zwischen IuK-technischer Infrastruktur und der Softwareentwicklung. Dann wird ein idealisierter Arbeitsprozess entworfen, der an die Gegebenheiten einer Softwareentwicklung und an bisherige Konzeptionen hinsichtlich des Softwareentwicklungsprozesses angepasst wurde. Die Teilprozesse dieses idealisierten Arbeitsprozesses, in dem die durch Normen bzw. Institutionen festgelegten Handlungen ablaufen, bestehen dabei aus Eintritt/Rekrutierung, Arbeitsplanung (Zieldefinition; Arbeitsverteilung von Arbeitspaketen mit detaillierter Entwicklungsplanung; Arbeitsverteilung von Posten), Leistungserstellung mit detaillierter Entwicklungsplanung und Arbeitszusammenführung.

Da das Arbeitsnetz ein Netzwerk ist, wird in Kapitel 3 der Stand der Forschung des Netzwerkdiskurses aufgerollt, der zugleich einen Organisationsformen- und Koordinationsdiskurs darstellt. Da Mischungen von Kooperation, Hierarchie/Macht und Wettbewerb als Koordinationsformen in Organisationsformen festgestellt wurden und der Fokus der Arbeit auf dem Arbeitsprozess der Softwareentwicklung auf der Handlungsebene liegt, werden die Organisations- und Koordinationsformen weitestgehend getrennt definiert. Dazu erfolgt eine systematische Auseinandersetzung mit Organisation/Organisationsformen und Koordination/Koordinationsformen, die ausgehend vom Institutionenbegriff bestimmt werden. Kooperation, Hierarchie/Macht und Wettbewerb als Koordinationsformen werden dazu anhand des Kategorisierungsschemas Handlungsbedingungen, Handlung und Ergebnis festgelegt.

Ausgehend von der Aufteilung der Softwareentwicklung in Teilprozesse und der Definition der Koordinationsformen, stehen im *zweiten Abschnitt* die Koordinationsformen im empirischen Feld im Zentrum. In Kapitel 4 wird dazu das empirische Vorgehen anhand der Datenerhebung, -aufbereitung und der Entwicklung von Indikatoren als Maß für die Institutionalisierungsstärke vorgestellt.

Das Open Source-Projekt KDE steht als Untersuchungsobjekt und Beispiel eines Arbeitsnetzes in Kapitel 5 im Mittelpunkt. Zuerst wird das Open Source-Projekt für ein besseres Verständnis im Rahmen einer Beschreibung der Open Source-Bewegung dargestellt. Akteure und IuK-technische Infrastruktur werden vorgestellt und danach die Ausprägungen

und Bedeutung der Koordinationsformen in den Teilprozessen der Softwareentwicklung analysiert.

Im *dritten und letzten Abschnitt* erfolgt in Kapitel 6 eine Analytik der vorgefundenen Koordinationsformen im Open Source-Projekt KDE. Anhand der Neuen Institutionenökonomie und des Neo-Institutionalismus werden Vermutungen theoretisch hypothetischer Art über die Koordination im Open Source-Projekt erstellt. Dabei werden die Theorien auf das KDE-Projekt angepasst und ihre Grenzen im Hinblick auf einen Schritt zu einer Theorie der Koordination in Organisationen aufgezeigt. Ausgangspunkt für eine Theoriemodifikation ist die Neue Institutionenökonomie in Kapitel 6.1. und der Neo-Institutionalismus in Kapitel 6.2., wobei beide Kapitel ähnlich strukturiert sind: Zuerst wird die jeweilige Theorie in Richtung des untersuchten Open Source-Projekts als informale Organisationsform modifiziert und vorgestellt, wobei erste Erkenntnisse über Softwareentwicklung und Arbeitsnetze (ggf. des Open Source-Projekts) einfließen. Danach werden Erklärungsansätze für die empirisch vorgefundenen Ausprägungen und Bedeutungen der Koordinationsformen in den Teilprozessen der Softwareentwicklung im Open Source-Projekt KDE anhand dieser Theorieinterpretationen geliefert.

In Kapitel 7 erfolgt die Zusammenfassung, wobei auch Anknüpfungspunkte für weitere Forschung aufgezeigt werden.

## **2. Virtuelle Arbeitsnetze als neue Arbeitsform: Arbeitsprozess der Softwareentwicklung und Rolle der Technik**

Ausgangspunkt der Untersuchung ist die Beobachtung (s. Kap. 2.1.), dass neue Arbeitsformen wie virtuelle Arbeitsnetze, darunter auch Open Source-Projekte, entstehen. Dieser Befund ist in folgendem Gesamtzusammenhang zu sehen: Traditionell wird die Erwerbsarbeit in Unternehmen untersucht, was auch bei informations- und kommunikationstechnisch vernetzter Arbeit nicht anders ist. Generell wird die Veränderung der Arbeitsweise durch vernetzte und vernetzende Informations- und Kommunikationstechnik (IuK-Technik) bei der Erwerbsarbeit oder die Beziehung zwischen Arbeitswelt und Lebenswelt bei Telearbeit analysiert (vgl. Huber, Hirschfelder 2004). In der Literatur wird zunehmend nicht mehr nur die Erwerbsarbeit betrachtet, die sich u.a. auch in verschiedene Formen ausdifferenziert, sondern auch weitere Arbeitsformen wie Ehrenamts, Eigen- und freiwilligen Arbeit wahrgenommen. Auch die Grenze zwischen Erwerbsarbeit und Nicht-Erwerbsarbeit verändert sich, wobei Erwerbsarbeit durch Nicht-Erwerbsarbeit und vice versa ersetzt wird. Somit kommen auch andere Arbeitsformen wie (virtuelle) Arbeitsnetze verstärkt in den wissenschaftlichen Fokus, die zum einen mit Erwerbsarbeit verbunden sein und zum anderen u.a. auch mit Ehrenamts, Eigen- und freiwilligen Arbeit zusammenhängen können.

Virtuell heißt in dem Fall, dass Kommunikation und Arbeit hauptsächlich über IuK-Technik erfolgen. Virtuelle Arbeitsnetze sind durch eine Zusammenarbeit autonomer, überwiegend individueller Akteure an einem Produkt mit überwiegend informalen Beziehungen gekennzeichnet und sind mit Telearbeit bzw. IuK-technisch vernetzter Arbeit verbunden. Ein Beispiel für ein virtuelles Arbeitsnetz ist ein Open Source-Projekt, dessen Arbeitsabläufe als unentgeltlich operierendes Arbeitsnetz nicht vollständig und abschließend analysiert wurden und daher Gegenstand dieser Arbeit sind. In beiden Fällen steht der virtuelle Arbeitsprozess („Workflow“) der Softwareentwicklung im Zentrum des Interesses, da wenig über die handlungsleitenden Normen/Regelungen im fast vollkommen virtuellen Arbeitsprozess bekannt ist.

Grundlegend für das Auftreten der virtuellen Arbeitsnetze ist die IuK-Technik, die nicht nur ein Produkt, sondern auch eine Infrastruktur darstellt. Daher wird die Frage nach dem Einfluss der IuK-Technik auf die Arbeit in virtuellen Arbeitsnetzen anhand einer technikphilosophischen und –soziologischen Perspektive reformuliert und präzisiert (s. Kap. 2.2.). Im Fokus steht dabei das Arbeitshandeln mit der Erstellung eines Produkts.

Da der Arbeitsprozess der Softwareentwicklung mit dem Arbeitshandeln im Mittelpunkt des Interesses liegt, wird dieser untersucht. Besonders interessieren die Normen, die das Arbeitshandeln strukturieren und in spezifischen, ähnlichen Situationen vorzufinden sind. Damit der Arbeitsprozess mit den darin auftretenden Normen analysiert werden kann, werden ähnliche, wiederholt auftretende Situationen als Teilprozesse gesucht und zusammengefasst. Es können dabei verschiedene Teilprozesse als Teile eines idealisierten Arbeitsprozesses abgegrenzt werden, die sowohl aus den Funktionen eines implizit angenommenen Arbeitsprozesses auf dem Arbeitsmarkt, als auch Phasenmodellen der Softwareentwicklung abgeleitet und zusammengesetzt werden (s. Kap. 2.3.). Arbeitsmarktfunktionen liegen wegen des Blicks auf den Arbeitsprozess und die Phasenmodelle wegen ihrer Anwendung in der Softwareentwicklungstheorie nahe. Zuerst werden als Softwareentwicklungsmodelle aus der Literatur das Wasserfallmodell, das evolutionäre Modell und zusätzlich das Open Source-Entwicklungsmodell vorgestellt. Danach werden die Teilprozesse als Teile eines idealisierten Arbeitsprozesses ausgehend von den Arbeitsmarktfunktionen entwickelt und dann mit Hilfe des Wasserfallmodells und evolutionären Modells angepasst und verfeinert, indem Übereinstimmungen und Detaillierungsmöglichkeiten ge-

sucht werden. Ein Open Source-Projekt steht im Mittelpunkt dieser Untersuchung, weswegen simultan das vorgestellte Open Source-Entwicklungsmodell anhand der verfeinerten, neu aufgestellten Teilprozesse auf seine Tauglichkeit für eine Untersuchung überprüft wird.

## **2.1. Virtuelle Arbeitsnetze als neue Arbeitsform**

In diesem Unterkapitel werden ausgehend vom Arbeitsbegriff und einer Betrachtung der gesellschaftlichen Arbeit virtuelle Arbeitsnetze als neue Arbeitsform festgestellt (s. Kap. 2.1.1.). Das virtuelle Arbeitsnetz wird dann definiert und die Unterform Open Source-Projekt als untersuchenswertes Beispiel eines virtuellen Arbeitsnetzes vorgestellt (s. Kap. 2.1.2.).

### **2.1.1. Virtuelle Arbeitsnetze - Zwischen Erwerbs-, Eigen-, freiwilliger und Ehrenamtsarbeit**

Virtuelle Arbeitsnetze und besonders Open Source-Projekte als Unterform der Arbeitsnetze verbreiten sich seit den 1990er Jahren immer mehr. Diese stehen zwischen virtueller Projektarbeit in und zwischen Unternehmen inklusive Erwerbsarbeit einerseits und virtuellen selbstorganisierten Gruppen im Internet inklusive freiwilliger Arbeit andererseits.

#### *2.1.1.1. Erwerbsarbeit als zentrale Untersuchungskategorie? – Neue Arbeitsformen im Fokus*

Der Fokus auf die Arbeit in der Gesellschaft, aber auch die Arbeit selbst z.B. als Erwerbsarbeit oder anderer Arbeitsformen, verändert sich. Neue Arbeitsformen wie (virtuelle) Arbeitsnetze entstehen. Dabei standen und stehen bestimmte Arbeitsformen im wissenschaftlichen Fokus, obwohl Arbeit eigentlich eine umfassende gesamtgesellschaftliche Erscheinung ist, die sich u.a. von Spiel oder Kunst unterscheidet (Hillmann 1994: 35f.). Dieser gesamtgesellschaftliche Charakter zeigt sich in folgender philosophischen Definition im philosophischen Wörterbuch von Schmidt (1991). Die Arbeit wird als etwas charakterisiert, das auf den Widerstand einer Sache stößt, wobei Kraft aufgewendet wird und der Sache „das Erstrebte“ ab- oder aufgerungen wird. Dieser allgemeine Begriff der Arbeit wurde durch den Begriff der Erwerbs-/Lohnarbeit überlagert, da er mit der Industrialisierung zur dominierenden gesellschaftlichen Arbeitsform wurde. Der bisher weitestgehend selbstbestimmte Arbeitsprozess wurde in einer z.B. unternehmerischen Organisation zu einer fremdbestimmten Arbeit (zur Geschichte der Erwerbsarbeit s. Eggebrecht et al. 1980).

Allgemein wurde der Begriff Arbeit bis Anfang der 1980er Jahre überwiegend mit Erwerbsarbeit gleichgesetzt. Auf dem Bamberger Soziologentag von 1982 wurde schließlich das Ende der Arbeitsgesellschaft im Sinne der Erwerbsarbeit vor dem Hintergrund der Dienstleistungsgesellschaft prognostiziert (Egloff 1996: 80). Seitdem wurden andere bzw. neue Arbeitsformen in eingeschränktem Rahmen (wieder-)entdeckt und untersucht, wie die Eigenarbeit im Haushalt bzw. in der Freizeit, ehrenamtliche Tätigkeiten (Ehrenamtsarbeit) und die Schwarzarbeit als nicht versteuerte Erwerbsarbeit (vgl. z.B. Heinze, Offe 1990; Hinrichs 1989; Wolff 1991).<sup>3</sup> Der wissenschaftliche Hauptfokus lag aber weiterhin auf der Erwerbsarbeit.

---

<sup>3</sup> Eigenarbeit umfasst die Arbeit, die ohne Entgelt von Individuen vor allem zur Reproduktion und im Haushalt geleistet wird. Ehrenamtsarbeit ist unentgeltliche Arbeit, die hauptsächlich in Form von formalen Posten in Vereinen erfolgt. Hinzuzählen kann man aber auch die unentgeltliche Arbeit in informellen Zusammenschlüssen z.B. bei der Organisation eines Nachbarschaftsfestes. Schwarzarbeit bezieht sich auf bezahlte Arbeit, bei der keine Steuern an den Fiskus abgeführt werden.

Neuere Entwicklungen zeigen, dass sich die Erwerbsarbeit und die Beziehung zwischen Erwerbsarbeit und Nicht-Erwerbsarbeit verändert und neue Arbeitsformen wie Arbeitsnetze entstehen. Dabei spielt u.a. die IuK-Technik eine Rolle. Der Fokus ist nicht nur auf verschiedene Formen der Erwerbsarbeit, sondern besonders auf weitere Arbeitsformen zu erweitern.

### 2.1.1.2. *Differenzierung und Diversifizierung der Arbeitsformen*

Von einer sehr stark dominierenden Erwerbsarbeitsform kann nicht mehr gesprochen werden, da eine quantitative Differenzierung der Beschäftigungsverhältnisse besteht. Der Anteil der Leiharbeiter, Beschäftigten in Arbeitsbeschaffungsmaßnahmen, sozialversicherungspflichtig Teilzeitbeschäftigten, befristet Beschäftigten, Scheinselbständige und geringfügig Beschäftigten/400-Euro-Jobs als eher atypische Arbeitsverträge nahm und nimmt zu, der Anteil der abhängig Beschäftigten in Normalarbeitsverhältnissen ab. Damit geht eine Flexibilisierung in der Erwerbsarbeit nicht nur mit einer Flexibilisierung der Erwerbsbiografie und einer Arbeitszeitflexibilisierung, sondern auch der Arbeitsinhalte einher, hin zu einem mehr an Projektarbeit (Miegel, Wahl 2002: 92ff.; Willke 1998: 82ff., 100ff., 146ff.; Schmiede 2006: 468ff.).

Zusätzlich bestehen Grenzveränderungen zwischen Erwerbsarbeit und Nicht-Erwerbsarbeit: Zum einen besteht eine Verschiebung vieler Einfach Tätigkeiten der Erwerbsarbeit in die Eigenarbeit, z.B. übernehmen Kunden bestimmte Tätigkeiten wie den Möbeltransport und –aufbau beim Möbelkauf (Miegel, Wahl 2002: 91f.; Voß, Rieder 2005; Wenny 1999: 205f.). Umgekehrt wurde weiblich dominierte haushaltsbezogene Arbeit seit den 1970er Jahren bis heute zu konsumorientierten Dienstleistungen umgewidmet, sprich Erwerbsarbeit, was z.B. bei der häuslichen Pflege zu beobachten ist (Häußermann, Siebel 1995: 175ff.). Ein und dieselbe Arbeit kann einerseits unbezahlt sein, andererseits jedoch bezahlt werden.

Zum anderen bestehen weiter Tendenzen zu diskontinuierlicher Beschäftigung oder einem Wechsel zwischen selbständiger und abhängiger Erwerbsarbeit. Neben einem Wechsel zwischen Erwerbs-, Eigen- und Ehrenamtsarbeit können diese auch parallel während einer Erwerbsbiografie auftreten (vgl. Willke 1998: 157f.; Mutz 2002; Schumacher 2003; Marrs, Boes 2003). Auch unterscheiden sich Eigenarbeit, ehrenamtliche Tätigkeiten und Schwarzarbeit eher wenig, da sie eher durch die sozialen Umstände bestimmt werden.

Bei einem wissenschaftlichen Blick auf gesellschaftliche Arbeit sollte daher nicht alleine von Erwerbsarbeit als zentraler Kategorie gesprochen, sondern ein übergeordneter, übergreifender Blick ausgehend von obiger eher philosophischer Begriffsdefinition eingenommen werden. (Gesellschaftliche) Arbeit wird dann ausgehend von diesem übergreifenden Blick anhand von verschiedenen Merkmalen wie bezahlt/unbezahlt, Flexibilisierungsgrad, Ort/Zeit der Arbeit (Haushalt, Unternehmen), Nutzung von IuK-Technik etc. bestimmt, der sich nicht ausschließlich auf die Erwerbsarbeit bezieht (vgl. Taylor 2004). Damit wird eine übergreifende Perspektive auf Arbeit gewählt, die nicht überwiegend die Erwerbsarbeit fokussiert, sondern auch andere und besonders neue Arbeitsformen in den Blick nimmt bzw. nehmen kann.

In diesem „Spannungsfeld“ zwischen verschiedenen Erwerbsarbeitsformen, Eigen- und Ehrenamtsarbeit können folgende Arbeitsformen auf diese Weise fokussiert werden:

Seit den 1970/80er Jahren verbreiteten sich beispielsweise neue Arbeitsformen wie Ein-Personen-Selbstständige oder Kleinstunternehmen im Rahmen eines Trends zu mehr Selbständigkeit, der aber mittlerweile nachgelassen hat. Der Anteil der Selbständigen an allen Erwerbstätigen stagniert seitdem (vgl. Leicht, Philipp 1999; Hoffmann, Walwei 1998;

Erlinghagen, Knuth 2002). Die Selbständigen arbeiten beispielsweise in abhängiger Erwerbsarbeit bzw. Scheinselbständigkeit und erstellen einerseits abgeschlossene Arbeitspakete als einzelne Selbständige, deren Auftrag Sie über das Internet erhalten haben. Andererseits arbeiten einige Selbständige in (Projekt-/Arbeits-)Netzwerken. Diese neuen Arbeitsformen liegen zwischen rechtlicher Selbständigkeit und ökonomischer Abhängigkeit (vgl. Reichwald et al. 2004; Wenny 1999; Leicht, Philipp 1999; Oertel et al. 1998: 351ff.; Apatzsch et al. 2005; Oertel et al. 2003; Borchert, Urspruch 2006; Sydow, Windeler 2004; Blair 2003).

Seit den 1990er Jahren beeinflusst die IuK-Technik die bisherige gesellschaftliche Arbeit mit ihren Formen und schafft zudem neue Arbeitsformen. Die IuK-Technik verändert mit ihrer zeitlichen und räumlichen Entgrenzung die Erwerbsarbeit, aber auch die Eigen- und Ehrenamtsarbeit. Arbeiten, die bisher in Betrieben vor Ort erledigt wurden, können durch die Technik über größere Entfernungen von Angestellten in anderen Betrieben des selben Unternehmens, über Unternehmensgrenzen hinweg oder von Selbständigen in virtueller Projektarbeit übernommen werden. Dabei kann die Arbeit, die sog. Telearbeit<sup>4</sup>, an wechselnden Orten, im eigenen Büro oder zu Hause auch von Privatpersonen/Ehrenamtlichen durchgeführt werden. Zusätzlich ist eine zeitliche Flexibilisierung möglich, da durch die IuK-Technik Informationen zwischengelagert und bei Bedarf abgerufen werden können (Schulz et al. 1999; Baukrowitz et al. 1998; Trautwein-Kalms 1997: 172).

Die IuK-Technik unterstützt das Entstehen von virtuellen (Projekt-/Arbeits-)Netzwerken aus Selbständigen. Damit stehen diese virtuellen Arbeitsnetze im Zusammenhang mit Erwerbsarbeit und Selbständigkeit (Reichwald et al. 1999: 461ff.; Reichwald et al. 2004: 51ff.; Wenny 1999: 165ff.). Diese virtuellen Selbständigen-Arbeitsnetze wurden auch im Zusammenhang mit virtuellen Unternehmen untersucht, die sich durch die IuK-technische Vernetzung mehrerer Unternehmen und durch das einheitliche Auftreten als Quasi-Unternehmung nach außen, jedoch bei gleichzeitiger rechtlicher Selbständigkeit der Unternehmen definieren (vgl. Arnold et al. 1995; Weigle, Krcmar 2000; Klatt 1999; Müller 1997; Büschken 1999). Ein Beispiel sind hoch virtuelle Selbständigen-Arbeitsnetze, die Software gegen Entgelt entwickeln (vgl. Henninger, Sieber 2001).

Zudem werden die Grenzen zwischen Erwerbsarbeit und Nicht-Erwerbsarbeit durch IuK-Technik durchlässiger. „Durch Computernetze werden die Grenzen von Arbeitsgruppen, Organisationen und Gemeinschaften häufig verwischt oder verschoben“ (Wellman 2000: 147; s.a. Picot et al. 1996). Softwarearbeiten von Freiwilligen können auch von Unternehmen verwendet werden oder Angestellte verschiedener Unternehmen arbeiten unentgeltlich zusammen, wobei deren Softwareprodukt dann in den Unternehmen genutzt wird. Solche Communities Of Practice oder Epistemic Communities treten als virtuelle, freiwillige, selbstorganisierte Gruppen im Internet in den letzten Jahren vermehrt auf und nehmen z.T. die Form von Open Source-Projekten an (vgl. Kap. 2.1.2.; sCohendet et. al. 2001; Wenger 1998; Edwards 2001; Schmiede 2006). In diesem Zusammenhang bildeten sich virtuelle selbstorganisierte Gruppen mit freiwilligen sozialen Beziehungen um gemeinsame Interessen, unterstützt durch textbasierte Interaktionen wie Foren, Usenet, MUDs/MOOs<sup>5</sup> oder Mailinglisten. Neben virtuellen Gruppen, die Geselligkeit als gemeinsames Interesse haben, bestehen auch Gruppen, die eine gemeinsame Leistungserstellung wie Softwareentwicklung verfolgen (vgl. Götzenburger, Löger 2000; Döring 2003: 489ff.). Diese freiwilligen, virtuellen Arbeitsgruppen sind Arbeitsnetze als neue Arbeitsform, die im Zusammen-

---

<sup>4</sup> Bei der Telearbeit arbeiten u.a. verschiedene räumlich getrennte Menschen virtuell zusammen, wobei je nach Arbeitsstandort zwischen Teleheimarbeit, Satelliten- oder Nachbarschaftsbüros und mobiler Telearbeit unterschieden wird.

<sup>5</sup> MUDs = Multi-User-Dungeons; MOOs = MUDs object oriented

hang mit Eigenarbeit und Ehrenamtsarbeit stehen. Ein Beispiel dieses Arbeitsnetztyps sind Open Source-Projekte, in denen freiwillig, größtenteils ohne Entgelt Software erstellt wird.

Virtuelle Arbeitsnetze außerhalb von Unternehmen, z.T. getragen von Einzelpersonen, sind eine neue Arbeitsform, die durch die IuK-Technik entstehen und bei denen die gemeinsame Arbeit z.B. im eigenen Haushalt oder eigenen kleinen Büro im Vordergrund steht (vgl. Brand 2004; Krömmelbein, Schmid 2000). Ihre temporären Arbeitszusammenhänge stehen dabei im Zusammenhang mit flexibler und alternierender Arbeit in Form von Arbeitsinhalten und abwechselnden Arbeitsformen in der Erwerbsbiografie (s.o.). Unterstützend wirkt die IuK-Technik, die eine örtlich und zeitlich ungebundene Arbeit ermöglicht.

Virtuelle Arbeitsnetze und virtuelle Projektarbeit ähneln sich durch den Einsatz von IuK-Technik und, da es sich um Gruppen handelt, im gruppendynamischen Modus des Arbeitsablaufs. Sie unterscheiden sich aber, indem sie in verschiedenen organisationellen Umwelten agieren. Virtuelle Projekte finden weitestgehend in den Unternehmensgrenzen statt, wenn auch oftmals an verteilten Standorten (Döring 2003).<sup>6</sup>

Weiterhin sind virtuelle Arbeitsnetze von (virtuellen) Rekrutierungsnetzwerken zu trennen, in denen Arbeitszusammenhänge durch freiwillige, informelle Beziehungen auch über eine Internet-Plattform angebahnt werden. Ein Arbeitsprozess mit der Erstellung eines Produkts findet nicht statt. Sie sind z.B. in der Medienbranche und besonders bei Fernseh- oder Filmproduktionen mit umfangreichen und temporären Projekte vorzufinden. Sie treten aber auch bei virtuellen Selbständigen wie Softwareentwicklern auf (vgl. Wenny 1999; Oertel et al. 2003; Windeler et al. 2000; Sydow, Windeler 2004; Blair 2003).

### **2.1.2. Open Source-Projekte als virtuelle Arbeitsnetze**

#### *2.1.2.1. Definition von (virtuellen) Arbeitsnetzen*

Unter einem Arbeitsnetz wird allgemein eine Verbindung bzw. ein Netzwerk von (rechtlich) autonomen Akteuren verstanden (s. Kap. 3 für allgemeine Netzwerkdefinition), die zusammen ein gemeinsames Produkt erstellen und vor allem auf informalen Beziehungen basieren, d.h. die Zusammenarbeit steht im Mittelpunkt dieser Definition. Arbeitsnetze treten einerseits mit geringer, aber auch hoher Virtualität auf. Bei virtuellen Arbeitsnetzen mit hoher Virtualität wird überwiegend vernetzte und vernetzende IuK-Technik bei Kommunikation und Produktion verwendet. Das gemeinsame Netzwerkprodukt ist im virtuellen Arbeitsnetz überwiegend ein informationsintensives Dienstleistungsprodukt wie ein Softwareprodukt, in anderen Fällen mit geringer Virtualität eher Dienstleistungen wie Film- bzw. Fernsehproduktion (z.B. Blair 2003; Sydow, Windeler 2004). Arbeitsnetze sind durch temporäre Arbeitszusammenhänge bestimmt, wobei Unterstützungsnetzwerke z.B. befreundeter Akteure wiederholt temporäre Arbeitszusammenhänge ermöglichen und damit das Arbeitsnetz stabilisieren. Temporäre Arbeitszusammenhänge und Unterstützungsnetzwerk können aber auch zusammenfallen. Kooperative Elemente wie Aushandlungsprozesse, Vertrauen, Reziprozität und gegenseitige Abhängigkeit können das Arbeitsnetz bestimmen. Die Verfolgung eines gemeinsamen Ziels unterstützt das gegenseitige Lernen, also die Innovationsfähigkeit, im Netzwerk. Die Produkterstellung erfolgt beim virtuellen Arbeitsnetz überwiegend zu Hause oder in kleinen, eigenen Büros, kann aber auch im nicht-virtuellen Arbeitsnetz vor Ort beim Auftraggeber erfolgen. Als autonome Akteure arbeiten

---

<sup>6</sup> Ein Vergleich mit der Projektarbeit in (Groß-)Unternehmen findet zwar in dieser Ausarbeitung nicht statt, kann aber, da die organisationellen Umwelten virtueller Arbeitsnetze thematisiert werden, durch Hinzuziehung des Lesers von weiterer Fachliteratur erfolgen.

idealerweise Einzelpersonen zusammen, d.h. überwiegend (Ein-Personen-)Selbständige oder Privatpersonen, da diese Akteure am ehesten ein Produkt zusammen erstellen (leicht verändert nach Brand 2004: 167ff.; s.a. Krömmelbein, Schmid 2000).

#### 2.1.2.2. *Open Source-Projekte als Arbeitsnetz*

In Arbeitsnetzen wird also die gemeinsame Arbeit im Netzwerk ins Zentrum gestellt, die sich z.B. als Softwareentwicklung in hoch virtuellen Arbeitsnetzen darstellt. Gemeinsame Arbeit für ein gemeinsames Produkt umschließt nicht nur Erwerbsarbeit, sondern auch andere Arbeitsformen wie Sie die Open Source-Projekte darstellen.

Im Folgenden werden Open Source-Projekte als Beispiele für die neue Arbeitsform Arbeitsnetz genauer betrachtet. Open Source-Projekte, die vorwiegend im Zusammenhang mit Eigenarbeit und Ehrenamtsarbeit diskutiert werden, sind Arbeitsnetze mit informalen Beziehungen zwischen Privatpersonen als individuelle, autonome Akteure. In einer gemeinsamen, freiwilligen Zusammenarbeit, der „peer production“<sup>7</sup>, wird ein gemeinsames Softwareprodukt als Netzwerkprodukt erstellt, das zugleich größtenteils ein öffentliches Gut darstellt. Kommunikation und Produktion erfolgen nur über IuK-Technik. Zudem sind sie auf der einen Seite von Dauer, was die anhaltende Wichtigkeit der Open Source-Projekte zeigt. Andererseits ist die informale Organisation durch seine Selbstorganisation recht flexibel. Das Projekt bietet also einen relativ stabilen Rahmen, in dem die temporäre Zusammenarbeit erfolgt.

Open Source-Projekte können ferner nicht als Markt angesehen werden, da kein Tausch von Gütern bzw. von Gütern gegen Geld inklusive Vertragsverhandlungen stattfindet.<sup>8 9</sup> Die Projekte sind keine bürokratischen, formalen Organisationen wie Unternehmen, die ein Produkt zur Vermarktung und Gewinnerzielung herstellen. Das Open Source-Projekt ist auch deswegen ein Arbeitsnetz, weil ähnliche Phänomene mit ähnlicher Funktionsweise in anderen Kontexten existieren, die eher als Arbeitsnetze identifizierbar sind.<sup>10</sup>

#### 2.1.2.3. *Arbeitsprozess und Einfluss der IuK-Technik*

Gerade beim Open Source-Projekt als Beispiel eines virtuellen Arbeitsnetzes interessiert wie der Arbeitsprozess abläuft und wie die IuK-Technik im Zusammenhang mit dem Arbeitsprozess eingesetzt wird. Beide Fragen wurden weder für Open Source-Projekte noch für virtuelle Arbeitsnetze als übergeordnete Begrifflichkeit annähernd abschließend und vollständig beantwortet, weshalb dies hier im Fokus steht.

---

<sup>7</sup> Anders als Demil und Lecocq (2003) identifiziert Benkler (2002) Open Source-Projekte mit der eigenständigen Form „peer production“, die unschwer als Netzwerk verstanden werden kann.

<sup>8</sup> Gleichwohl wird gerne die Marktmetapher zur Beschreibung des Open Source-Projekts herangezogen (s. z.B. Ghosh 1998).

<sup>9</sup> Zusätzlich ist es eher schwierig, Software als Erfahrungs-/Wissensgut in Märkten zu erstellen, da in Märkten nur wenige Informationen z.B. nur Qualität und Preis (auch wegen des Informationsparadoxons, der nicht mehr nötigen Bezahlung nachdem eine Information preisgegeben wurde) sowie kaum zukünftige Entwicklungen kommuniziert werden (Powell 1996: 223).

<sup>10</sup> Das Open Source-Projekt ähnelt der virtuellen Zusammenarbeit von selbständigen Unternehmensberatern, da auch in diesem autonome Akteure auf internationaler Ebene zusammenarbeiten und hauptsächlich Emails und Telefonkonferenzen nutzen. Die selbständigen Unternehmensberater arbeiten zwar nicht an Software, aber unentgeltlich an einer neuen Standardisierung für ihre Profession als Netzwerkprodukt. Sie arbeiten, wie die Open Source-Mitglieder, nicht in ihrer Funktion als Selbständige zusammen, sondern als Privatpersonen, wobei diese Arbeit jedoch Spill-overs auf die Auftragslage und die Qualität ihrer Erwerbsarbeit hat (vgl. Brand, Schmid 2003). Auch die Funktionsweise ähnelt dem Open Source-Projekt: Ausgehend von einer grundlegenden Kooperation besteht eine Hierarchie zwischen einer Steuerungsgruppe mit hoher Reputation und diversen modularen Subprojekten. Ein Wettbewerb erfolgt nur um Reputation (vgl. Brand, Schmid 2003e).

Der Arbeitsprozess in Arbeitsnetzen wurde bisher eher vernachlässigt, weswegen in dieser Untersuchung der Ablauf von Arbeitsprozessen und deren Verregelung bzw. Normierung fokussiert wird. Arbeitsnetze werden zwar im Rahmen der Erwerbsarbeit und der Projektarbeit von Selbständigen betrachtet (s.o.), doch liegt der Hauptfokus weiterhin auf der Erwerbsarbeit im Unternehmen. In diesem Zusammenhang wird bei der Erwerbsarbeit besonders die veränderte Arbeitsweise durch IuK-Technik oder die Beziehung zwischen Arbeitswelt und Lebenswelt bei Telearbeit analysiert (vgl. Huber, Hirschfelder 2004; Baukowitz et al. 2006; Goll 2002; Boos et al. 2000; für angestrebte Forschungsfoki s. Boes et al. 2006).

Virtuelle Arbeitsnetze werden auch als Communities Of Practice betrachtet, in denen eine praktische Kooperation, Netzwerkformen, Nutzung von IuK-Techniken, Wissenstransfer und Arbeit zusammen auftreten. „Bislang liegt jedoch nur eine begrenzte Zahl von Untersuchungen zu Arbeitsprozessen vor; viele Untersuchungen beziehen sich auf lokale *communities*“ (Hervorh. im Original; Schmiede 2006: 468).

Von besonderem Interesse sind virtuelle Arbeitsnetze, in denen die IuK-Technik ständig und fast ausschließlich bei Produktion und Kommunikation verwendet wird. Diese hohe Virtualität kann besonders bei der Erstellung virtueller Produkte wie Software angetroffen werden, da die Softwareerstellung direkt mit der IuK-Technik zusammenhängt (s. Kap. 2.2.). Zudem ist die Softwareentwicklung in virtuellen Arbeitsnetzen interessant, da in der Informatik die Softwareentwicklung vor allem in Unternehmen analysiert und mit Phasenmodellen beschrieben wird. Es werden normative Vorstellungen entwickelt, wie ein Softwareentwicklungsprozess abläuft bzw. ablaufen soll (s. z.B. Friedrich 1992; Sommerville 2001; Balzert 1996; Waterson et al. 1997).

Die Phasenmodelle sind Vorgehensmodelle und damit Vorstellungen eines idealen Arbeitsprozesses bei der Softwareentwicklung, wobei der zeitliche Ablauf im Vordergrund steht. Sie können aber durch die zeitliche Phasenabfolge der Komplexität der Softwareentwicklung nicht gerecht werden. Dies zeigt sich daran, dass in Unternehmen Phasenmodelle zwar zur Orientierung genutzt, diese aber situativ angepasst werden (Welz, Ortman 1992: 81f.). Daher sind die Phasenmodelle analytischer Natur und sie bilden zugleich wichtige Bereiche des Arbeitsprozesses ab. Sie können vielmehr für die Entwicklung eines idealisierten Arbeitsprozesses mit Teilprozessen verwendet werden (s. Kap. 2.3.). Folglich wird bei Nutzungs-/Arbeitsstilen, Projekttypen und Phasenmodellen nicht untersucht, wie und an welchen Punkten Handlungsnormen während der Softwareentwicklung Anwendung finden.

Auch bei den sozialwissenschaftlich orientierten empirischen Untersuchungen wird die Softwareentwicklung in Unternehmen unter dem Aspekt der Phasenmodelle, aber auch Arbeitsstile und Projekttypen untersucht (s. z.B. Welz, Ortman 1992; Konrad, Gerd 1999; Bittner et al. 1995; Henninger, Sieber 2001). Im eher technisch geprägten Informatik-Forschungsbereich der Computer Supported Cooperative Work (CSCW) werden u.a. auch virtuelle Arbeitsnetze, jedoch nicht unbedingt der Softwareentwicklung untersucht (z.B. Törpel et al. 2002; Mark 2002). Der Hauptfokus dieses Forschungsbereichs liegt auf der Frage, wie die IuK-Technik im Rahmen der Mensch-Maschine-Interaktion an die Erfordernisse der Zusammenarbeit angepasst werden kann (Döring 2003: 284; s. z.B. International Journal on CSCW).

Hoch virtuelle Arbeitsnetze als eigene Arbeitsform wurden also bisher, außer von Becher et al. 2005, wenig untersucht. Die Autoren betrachten die Softwareentwicklung also entweder auf einer individuellen Ebene in Form von IuK-technischen Nutzungs- bzw. Arbeitsstilen oder auf einer umfassenden (Projekt-)Gesamtebene der Softwareentwicklung in Form von Projekttypen und Phasenmodellen. Nicht-Erwerbsarbeit auch virtuellen Charak-

ters wird eher am Rande behandelt. Es bestehen kaum quantitative Studien über ihre Verbreitung.

Als ein besonderes, hochvirtuelles Arbeitsnetz, in dem überwiegend unentgeltlich gearbeitet wird, wecken Open Source-Projekte und besonders deren Arbeitsprozesse mit den beeinflussenden Normen Interesse. Der Arbeitsprozess mit seinen Normen und Regeln in einem Open Source-Projekt als beispielhaftes virtuelles Arbeitsnetz wird daher in dieser Arbeit untersucht.

Dieses Arbeitsnetz wurde in den Sozial- und Wirtschaftswissenschaften und auch in der Informatik seit Ende der 1990er Jahre verstärkt fokussiert. In den Diskursen wurden zwar allgemeingültige Normen formuliert, jedoch bisher die Anwendung dieser Normen bzw. Institutionen im Arbeitsprozess nicht thematisiert (s. Kap. 5.4.1.). Spezielle Normen werden in bestimmten, ähnlichen Situationen angewendet. Die oben genannten Phasenmodelle versuchen beispielsweise den Arbeitsprozess der Softwareentwicklung in ähnlichen Situationen zu fassen, daher muss sich damit auseinandergesetzt werden. Der Arbeitsprozess steht deshalb, wie einleitend erwähnt, in Kapitel 2.3. im Mittelpunkt, um ähnliche, wiederholt auftretende Situationen festlegen zu können. Diese ähnlichen Situationen können als Untersuchungsraster und zur Kategorisierung des Arbeitsprozesses dienen, mit deren Hilfe dieser genauer untersucht werden kann.

Zusätzlich spielt im Open Source-Projekt, da fast nur über IuK-Technik produziert und kommuniziert wird, wie oben angedeutet, die IuK-Technik eine besondere Rolle. Da die Softwareentwicklung und der Arbeitsprozess ohne Betrachtung der IuK-Technik wegen der hohen Virtualität nicht verstanden werden kann, wird die IuK-Technik im nächsten Kapitel 2.2. betrachtet. Dabei wird auf das Zusammenspiel zwischen IuK-Technik und Arbeitsprozess fokussiert.

## **2.2. IuK-Technik im Arbeitsprozess virtueller Arbeitsnetze**

Die vernetzte und vernetzende Informations- und Kommunikationstechnik (IuK-Technik) ist eine Grundlage für die Existenz von hoch virtuellen Arbeitsnetzen der Softwareentwicklung. Daher ist die Frage interessant, wie diese Technik die Arbeit bzw. den Arbeitsprozess z.B. in einem virtuellen Arbeitsnetz beeinflusst. Technik wurde bisher in einer weiten Definition der Sozial- und Geisteswissenschaften als Zweck-Mittel-Verhältnis oder in einer engen Definition der Technik-/Naturwissenschaften als Artefakt definiert. Beide Definitionen werden aber der Technik nicht vollends gerecht, weswegen Ropohl (2001: 17) einen Technikbegriff mit mittlerer Reichweite vorschlägt. Er definiert Technik und damit auch die IuK-Technik als materielle Artefakte oder Sachsysteme von Gesellschaften, die in Abgrenzung zur ersten Natur von Menschen künstlich hergestellt und zugleich von diesen genutzt werden.

Zur IuK-Technik gehört das Internet mit seiner Hard- und Software, die Computer von Internetnutzern als Hardware ebenso wie deren verwendete Software. Die IuK-Technik der virtuellen Arbeitsnetze ist eine Teilmenge dieser Hard- und Software. Auf den Großteil des Internets und der Computer der Nutzer kann im Rahmen der Arbeitsnetze kaum zugegriffen werden. Einerseits werden in den untersuchten Open Source-Projekten Software erstellt, andererseits besteht am ehesten Zugriff auf die Software als Infrastruktur. Da also Software zugleich ein Produkt, aber auch eine Infrastruktur darstellt, wird die Rolle dieser Technik aus technikphilosophischer und –soziologischer Perspektive fokussiert. Erstens wird die IuK-Technik als statische, gekapselte Struktur (Black Box bzw. Gestell) betrachtet (s. Kap. 2.2.1.). Zweitens ist die Technik dadurch ein Medium der Welterschließung (s. Kap. 2.2.2.). Drittens rückt die Produktion der IuK-Technik in Form der Software in den

Vordergrund, wobei ein Einfluss verschiedener Akteure besteht (s. Kap. 2.2.3.). Dabei wird die oben genannte Frage reformuliert und präzisiert.

### **2.2.1. IuK-Technik als Gestell bzw. als Black Box**

IuK-Technik kann nach Heidegger (1982) als ein Gestell betrachtet werden, d.h. Gestell definiert Werkzeuge und Maschinen wie Internet oder Computer aber auch (Open Source-) Softwareprodukte, an denen sich die Menschen aufrichten, die das Leben vereinfachen und dem sie als etwas von außen Gegebenes begegnen. Diese Artefakte sind mit Vorstellungen von Technik im Sinne eines Zweck-Mittel-Verhältnisses verbunden (Heidegger 1982: 6). Ein Gestell ist im Falle der IuK-Technik die stabilisierte Infrastruktur aus Software und Hardware, die eine Produktion und Kommunikation im Internet mit seinen Computern ermöglicht. Die Immaterialität der Software ist zwar ersichtlich, baut aber auf physischen Artefakten (Hardware, elektrische Ströme) auf, die auf einer übergeordneten Ebene manipuliert werden.

Bei Software, auch bei Open Source-Software, besteht ein Unterschied zwischen textbasiertem Quellcode und dem kompilierten und laufenden Maschinencode, wobei der Maschinencode als Artefakt auftritt. Der Maschinencode ist als statisches Artefakt nach der „Actor-Network Theory“ eine gekapselte Black Box, d.h. er ist durch die Kompilierung nicht einsehbar und unverständlich, wogegen der Quellcode lesbar und veränderbar ist (vgl. Tuomi 2001). Die Software basiert auf einer Zweck-Mittel-Beziehung, die von menschlichen Akteuren, in diesem Fall den Projektmitgliedern, gesetzt wird und funktional organisiert ist, d.h. aus Algorithmen besteht (vgl. Hubig 2004; Heintz 1993: 234ff.). Die Algorithmenisierung mit der vom Menschen gesetzten Zweck-Mittel-Beziehung ist als Maschine aufzufassen. Damit ist der Open Source-Maschinencode als Black Box eine immaterielle Maschine, die auf Hardware basiert. Der textbasierte Quellcode kann dagegen nicht direkt für einen Zweck eingesetzt werden, weswegen dieser nicht als Technik anzusehen ist. Seinen Zweck erreicht der Quellcode erst als Maschinencode im gekapselten Black Box-Zustand.

Drei Argumentationen umschreiben die Nutzung der immateriellen Maschine als Zweck-Mittel-Verhältnis. Erstens ist die Technik wertfrei, d.h. könne für jeden Zweck verwendet werden. Zweitens ist der Technik durch ihre Produktion ein Zweck innewohnend und Macht sowie Herrschaft werden dadurch etabliert (vgl. Marcuse 1994). Drittens besteht in der Literatur oftmals ein deterministischer Blick, dass Technik Organisationsstrukturen kausal bestimmt (vgl. Orlikowski 1992). Alle drei Argumentationen verengen ihren Blick auf einzelne Sachverhalte, die sich aber nicht ausschließen. Vielmehr gelten alle drei Argumentationen zugleich. Der Umgang mit Artefakten z.B. der IuK-Technik ist nicht determiniert, aber bestimmte Verwendungsweisen mit Zweck-Mittel-Beziehung sind intendiert. Bestimmte Techniken wie die IuK-Technik sind auf einen spezifischen, eng eingegrenzten Verwendungszweck hin entwickelt worden, so dass der Bereich der alternativen Verwendungsmöglichkeiten eher gering ist. Damit liegt die Frage nach dem Entstehungszusammenhang und den beteiligten Akteuren nahe. Dabei ist Technik das Ergebnis von strategischer Wahl und sozialer Interaktion und wirkt wiederum rekursiv auf die Technik (vgl. Orlikowski 1992).

### **2.2.2. IuK-Technik als Medium**

Die IuK-Technik im Open Source-Projekt ist nicht nur eine Maschine, sondern auch ein Medium, das zur Welterschließung dient (vgl. Esposito 1993; Rammert 2000, 115ff.; Hubig 2004). Dabei wird die Technik, im vorliegenden Fall die Software und die Hardware der Infrastruktur, als Black Box durch die gekapselte, statische Zweck-Mittel-Beziehung zur

Welterschließung verwendet. Damit unterscheidet sich die luK-Technik aber nicht von anderer Technik (vgl. Hubig 2004).

Die Grundlage des Mediums bildet die vernetzte, textbasierte, computerbezogene luK-Technik des Internets zur digitalen Datenübertragung. Die Technik ermöglicht das Umwandeln von analogen Daten in ein für Computertechniken lesbares Format und die Verarbeitung sowie Übermittlung von Informationen in Sprache, Bild und Ton über ein und dasselbe Medium bei gleichzeitiger Zwei-Wege-Kommunikation (vgl. Krömmelbein 2004; Malone et al. 1987: 488; Kollmann 2001: 26; Döring 2003: 37ff.).<sup>11</sup>

Die Anwender dieser luK-Technik können sich die Welt durch Kommunikation und Interaktion mit anderen Anwendern erschließen, wobei jeder mit Zugang z.B. zum Internet dieses nutzen kann. Damit geht die Problematik der Anonymität im Internet einher. Jeder Techniknutzer hat zwar eine identifizierbare Adresse, doch ist eine digitale Signatur oder eine Email-Adresse nicht untrennbar mit einer Person verbunden und kann daher von verschiedenen Personen verwendet werden (vgl. Kumbruck 2000).<sup>12</sup> Auf der anderen Seite hat sich eine Kultur der Ehrlichkeit im Internet („Netiquette“) etabliert (vgl. Döring 2003: 283).

Zusätzlich wird die Kommunikation und Interaktion durch die Besonderheiten des Mediums bestimmt. „Die Reduktion räumlicher Schranken und Beschleunigung von Interaktionen, die Möglichkeit von synchroner und asynchroner Kommunikation, die Dominanz der textbasierten Kommunikation, ihre Interaktivität, und die Kopplung und Verzahnung unterschiedlicher Interaktions- und Kommunikationsformen und –räume“ sind als Kerncharakteristika dieses Mediums zu verstehen (vgl. Krömmelbein, Schmid 2004: 214). Die Interaktivität zeichnet sich durch eine one-to-many- und many-to-one- neben einer one-to-one-Kommunikation mit Push- und Pull-Strategien des Informationsversendens und –erhalts aus. Die räumlich und zeitlich synchrone und asynchrone Kommunikation und Produktion (anytime-anyplace) ermöglicht dabei eine räumliche und zeitliche Entkopplung der Techniknutzer (vgl. Kollmann 2001: 27; Runkehl 1998; Döring 2003: 37ff.).

Durch diese Charakteristika wird die Integration bisher getrennter Arbeitsvorgänge der Planung, Abstimmung und des Kommunikationsaustauschs z.B. über ein textbasiertes Medium ermöglicht. Dies gilt für Unternehmen genauso wie für Arbeitsnetze z.B. in Form von Open Source-Projekten, Selbständigen-Arbeitsnetzen oder anderen Netzwerkformen, die dadurch aber erst ins Leben gerufen werden (Krömmelbein, Schmid 2004: 215). In Softwareprojekten führt dies zu einer gemeinsamen, gleichzeitigen und parallelen Bearbeitung von Software.

### **2.2.3. Zusammenhang zwischen luK-technischer Infrastruktur und Softwareentwicklung**

Die luK-Technik ist ein Gestell, eine gekapselte Black Box bzw. ein Zweck-Mittel-Verhältnis, das zugleich als Medium zur Welterschließung genutzt wird. Die luK-Technik als gekapselte Black Box unterstützt aber nicht nur die Interaktion und Kommunikation, sondern auch die Produktion. Das Wesen der Maschine ist innig mit der Form des Mediums verbunden, d.h. es besteht eine direkte und unmittelbare Verbindung zwischen Ma-

---

<sup>11</sup> Dies wird auch als „Technological Integration Effect“ oder „Technologieparadigma“ bezeichnet.

<sup>12</sup> Auch wenn biometrische Muster als Signaturen verwendet werden, können diese unkontrolliert von anderen Personen verwendet werden, da sie ins digitale Format übertragen werden müssen und damit von der Person entäußert werden (vgl. Kumbruck 2000). Hohe Transaktionskosten der Kopieerstellung oder einschneidende Sanktionen können dieses Problem zusätzlich mildern.

schine und Medium sowohl auf der Software- als auch auf der Hardwareebene. Dies gilt auch für die Produktion.

Dieser Blick auf die Nutzung vernachlässigt aber die besonderen Umstände der Produktion. Einerseits wird bestehende Technik im Sinne ihres Zweck-Mittel-Verhältnisses für eine Produktion verwendet. Andererseits wird die Technik selbst auch erstellt. Die Werkzeuge als IuK-technische Infrastruktur und Medienbestandteile können also wiederum selbst entwickelt und produziert werden. Dabei werden Artefakte bei der Entwicklung von Artefakten verwendet. Wenn also ein Artefakt produziert wird, ist diese gekapselte Black Box „geöffnet“, aber gleichzeitig werden andere gekapselte Artefakte als Medium für die Herstellung verwendet. Diese rekursive Natur der Technologie zeigt sich in den statischen und dynamischen Elementen bei der Technikentwicklung („Dualität der Technologie“: s. Orlikowski 1992).<sup>13</sup> Zugleich ermöglichen die gekapselten Artefakte eine Komplexitätsreduktion. Die Erkenntnis der aufeinander angewendeten Artefakte basiert auf den wissenschaftstheoretischen Laborstudien, die mit der Actor-Network Theory untersuchen, wie Wissen generiert und mit Technik umgegangen wird. Dabei wird z.B. in der Physik beobachtet, an welchen Punkten eines Forschungsprojekts die gekapselten Artefakte eine Rolle spielen, welche direkt beteiligten Akteure dabei wie agieren und nach welchen Regeln die Produktion erfolgt (vgl. Latour, Woolgar 1979; Tuomi 2001). Damit sollte die Wechselwirkung zwischen Softwareerstellung und den verwendeten gekapselten Artefakten im Fokus stehen.

Bei den Arbeitsnetzen ist daher die produzierte Software von der IuK-technischen Infrastruktur als Medium zu trennen, auf deren Basis die Produktion und Kommunikation erst erfolgen kann. Die IuK-technische Infrastruktur setzt sich aus den Computern der Arbeitsnetzmitglieder, der hardware- oder softwaretechnischen Infrastruktur des Arbeitsnetzes und des restlichen Internets zusammen. Im Fokus der Untersuchung ist die IuK-technische Infrastruktur des Arbeitsnetzes, die die Produktion und Kommunikation in einem Arbeitsnetz unterstützt und auf die vom Arbeitsnetz oder seinen Beteiligten größtenteils zugegriffen werden kann. Zusätzlich ist zwischen der Software als bearbeiteter Quellcode und der unterstützend genutzten, black-boxed Software als IuK-technische Infrastruktur zu unterscheiden (s. Kap. 2.2.1.)

Das Besondere an der IuK-Technik und speziell der Software ist aber, dass durch die oben genannten Zusammenhänge eine direkte und unmittelbare Anreicherung von Wissen durch neues Wissen ohne Umwege erfolgen kann, d.h. auf bestehender Arbeit inkrementell aufgebaut wird bzw. die gleiche Technik auf sich selbst angewendet wird, so dass eine Rekursion bzw. Wissensspirale entsteht (vgl. Castells 1996).

„Die eingesetzten IuK – Techniken haben nicht Werkzeugcharakter zur Unterstützung außer ihnen liegender Aufgaben, sondern sie sind Bestandteil eines Gesamtprozesses, eines Systems: Innovationen werden generiert und in einem kumulativen Rückkopplungszusammenhang wieder für Innovationen genutzt“ (Schmiede 2003: 176).

Dies ist möglich durch die Bearbeitung des offenen, nicht black-boxed Quellcodes bei gleichzeitiger Nutzung des bisher geschriebenen, aber kompilierten Maschinencodes für die Infrastruktur. Dabei kann zwischen einer unverzögerten Wissensspirale, d.h. Anwendung von Wissen auf Wissen, bis hin zu einer verzögerten Wissensspirale unterschieden werden, bei der weiteres Wissen erarbeitet bzw. herangezogen werden muss.

---

<sup>13</sup> Orlikowski (1992) schlägt daher ein strukturalistisches Modell vor, das hier aber nicht weiter verfolgt wird.

Bei der Softwareentwicklung mit der genutzten IuK-technischen Infrastruktur und Wissensspirale spielen nach der Actor-Network Theory und den wissenschaftstheoretischen Laborstudien (s.o.) die individuellen Akteure eine Rolle. Der Entstehungszusammenhang und die beteiligten Akteure nehmen u.a. Einfluss, da nach obiger Erkenntnis (s. Kap. 2.2.1.) der Umgang mit Artefakten oder Technik nicht determiniert ist, aber bestimmte Verwendungsweisen intendiert. Damit determiniert die verwendete IuK-technische Infrastruktur als Artefakt die Technikentwicklung nicht. Somit wird der von der Actor-Network Theory betrachtete Forschungsprozess, der bei den Arbeitsnetzen die Softwareentwicklung ist, als Arbeitsprozess betrachtet. Daraus lässt sich ableiten, dass die IuK-technische Infrastrukturtechnik der Arbeitsnetze wie des Open Source-Projekts keinen determinierenden Einfluss auf die Normen bei der Softwareentwicklung hat, aber bestimmte Normen unterstützen kann. Die einflussnehmenden, individuellen Akteure sind in Arbeitsnetzen deren Mitglieder, z.B. Softwareentwickler. Weitere Einflüsse auf die Softwareentwicklung als Technikentwicklung gehen laut historischen Sozialkonstruktivisten von gesellschaftlichen Interessensgruppen, besonders den Endnutzern, vor allem aus der Umwelt der Produktion aus (vgl. Pinch, Bijker 1987).<sup>14</sup> Das ingenieurwissenschaftliche Konstruieren, damit auch die Softwareentwicklung, ist in soziale Beziehungen eingebettet, wobei der Konstrukteur/Softwareentwickler die Interessen verschiedener gesellschaftlicher Gruppen mit betrachtet (vgl. Glock 1997). Dabei werden besonders die Endnutzer mit in die Untersuchung der Softwareentwicklung einbezogen, da diese mit dem erstellten Endprodukt direkt und damit mit der Softwareentwicklung in Verbindung stehen. Weitere Interessensgruppen haben zwar einen Einfluss auf die Softwareentwicklung und das Produkt, werden aber vernachlässigt, da außer den Endnutzern andere Interessensgruppen nur einen indirekten Einfluss haben. Da auf Open Source-Projekten ein Hauptfokus liegt, werden Einflüsse von Interessensgruppen im Rahmen der neo-institutionalistischen Analyse in Kapitel 7.2. untersucht.

Endnutzer sind die Nutzer, die das produzierte Produkt weitestgehend ausschließlich nutzen. Da Endnutzer mittlerweile Teilarbeiten ausführen, die vorher im Rahmen von Erwerbsarbeit durchgeführt wurden (s.o.), kann sich die Grenze zwischen Produzenten und Konsumenten/Nutzern ein Stück weit auflösen bzw. verschieben. Die Wissensspirale bezieht sich also auf die Softwareentwicklung als Arbeitsprozess, bei der die IuK-technische Infrastruktur den Arbeitsprozess unterstützt und die Technik generierenden Softwareentwickler sowie (End-)Nutzer in den Arbeitsprozess einbezogen sind.

Zusammenfassend besteht also ein Zusammenhang, wenn nicht eine Wechselwirkung zwischen IuK-technischer Infrastruktur und Softwareentwicklung, da die Technik gleichzeitig erstellt wird und unterstützend wirkt. Daher ist eher zu Fragen, wie die stabilisierte, gekapselte IuK-technische Infrastruktur den Arbeitsprozess und seine Normen<sup>15</sup>, d.h. die Produktion von neuem Quellcode, in den Arbeitsnetzen unterstützt bzw. wie die Technik genutzt wird. Der Arbeitsprozess der Softwareentwicklung wird, wie oben argumentiert, von den Mitgliedern der Arbeitsnetze, aber auch den Endnutzern als Akteuren beeinflusst und vorher dargestellt. Da eine Verwendung von Artefakten auf Artefakte in Form von IuK-technischer Infrastruktur auf die Softwareentwicklung erfolgt, wird auch die gekapselte IuK-technische Infrastruktur vorher vorgestellt. Die Technik unterstützt einerseits als Medium die Information und Kommunikation im Arbeitsprozess, andererseits die Produktion.

---

<sup>14</sup> Der Einfluss von gesellschaftlichen Gruppen, betrachtet durch die Linse des soziologischen Neo-Institutionalismus, steht in Kapitel 7.2 im Fokus. Dabei bestehen theoretische Anknüpfungspunkte für die historischen Sozialkonstruktivisten wie Pinch und Bijker (1987).

<sup>15</sup> Als Normen werden die Koordinationsformen Kooperation, Hierarchie/Macht und Wettbewerb untersucht (s. Kap. 3.).

Daher wird die verwendete IuK-technische Infrastruktur nach ihrer Kommunikations-, Informations- und Produktionsausrichtung eingeteilt.

### **2.3. Entwicklung eines idealisierten Arbeitsprozesses als Analyserahmen für die Softwareentwicklung**

Bisher wurde die Beziehung zwischen virtuellem Arbeitsprozess und IuK-Technik beschrieben, die eine Bedingung für den Arbeitsprozess ist. In diesem Kapitel steht der virtuelle Arbeitsprozess der Softwareentwicklung als „Workflow“ in Arbeitsnetzen im Mittelpunkt der Untersuchung. Damit ist auch eine Transformation von Arbeitskraft in Arbeitsleistung verbunden, die nicht nur bei Erwerbs- sondern auch Nicht-Erwerbsarbeit auftreten kann (s. Kap. 2.1.). Es sollen die strukturierenden Normen bzw. Institutionen der Softwareentwicklung untersucht werden, d.h. die Frage, wie Normen Handlungen im ablaufenden Arbeitsprozess bestimmen. Die Normen beziehen sich auf spezifische Situationen, in denen diese angewendet werden. Für eine Untersuchung des Arbeitsprozesses ist daher eine punktuelle bzw. situative Unterteilung zu finden, die sich in Form von Teilprozessen des Gesamtarbeitsprozesses darstellen lässt, in denen auch Normen einen Einfluss haben. Die Teilprozesse können sowohl als Funktionen nach ihrem Zweck für den Gesamtprozess als auch als zeitliche Phasenabfolge von Arbeitsabläufen betrachtet werden. Funktionen scheinen aber nicht in einem Zusammenhang zueinander zu stehen und ihre Benennung anhand ihres Zwecks wird durch ein spezifisches Forschungsinteresse bestimmt. Bei Phasenabfolgen werden z.T. parallel stattfindende Arbeitsabläufe wie beim Open Source-Entwicklungsmodell übersehen bzw. Teile der Arbeitsabläufe können unpassend zugeordnet werden. Um alle Bereiche des Arbeitsprozesses abzudecken und die jeweiligen Probleme einer funktionalen und zeitlichen Sicht auszugleichen, wird ein idealisierter Gesamtprozess entwickelt. Dieser idealisierte Gesamtarbeitsprozess stellt dabei den Analyserahmen für die Untersuchung der Arbeitsnetze dar, in dem die strukturierenden Normen identifiziert werden.

Für die funktionale Sicht können die Arbeitsmarktfunktionen herangezogen werden, die im Rahmen einer ökonomischen Analyse des Arbeitsmarkts einen implizit angenommenen Arbeitsprozess untersuchen. Die Arbeitsmarktfunktionen schließen dabei an den Fokus dieser Untersuchung auf die Zusammenarbeit und das Arbeitshandeln vor allem in Unternehmen an. Bei der zeitlichen Sicht bieten sich Phasenmodelle der Softwareentwicklung aus der Informatik an, die aber bisher weitestgehend die Softwareentwicklung in Unternehmen abbilden. Die verschiedenen Phasenmodelle der Softwareentwicklung zeigen aber einen differenzierten Blick auf den Softwareentwicklungsprozess als Arbeitsprozess. Beide Ansätze bieten sich für eine Betrachtung der Softwareentwicklung in den virtuellen Arbeitsnetzen inklusive Open Source-Projekten an.

Aus diesen beiden Blickwinkeln wird daher ein idealisierter Arbeitsprozess als Analyserahmen entwickelt. Dabei wird auf Gemeinsamkeiten und Unterschiede geachtet, wobei unterschiedliche Teilbereiche, soweit diese sinnvoll sind, in den Analyserahmen aufgenommen werden. Bei den Arbeitsmarktfunktionen und den Phasenmodellen der Softwareentwicklung steht zwar eher Erwerbsarbeit im Rahmen von Unternehmen im Fokus, doch wird davon ausgegangen, dass Teile der Arbeitsmarktfunktionen und Phasenmodelle auch für Nicht-Erwerbsarbeit und Arbeitsnetze verwendet werden können. Wie oben in der philosophischen Definition (in Kap. 2.1.) dargestellt, führt jede Person eine Transformation von Arbeitskraft in Arbeitsleistung als Arbeitsprozess mittels des Einsatzes ihrer Person und ihrer Ressourcen durch. Die Entlohnung einer Arbeit sowie die Arbeitserstellung in einer Organisation sind beides Dimensionen, die den Arbeitsprozess mit seinen Normen beeinflussen, jedoch den Arbeitsprozess als solchen nicht wesentlich verändern. Der Ana-

lyserahmen wird daher so abstrakt wie möglich gehalten, dass damit Arbeitsnetze vor allem im Kontext mit Nicht-Erwerbsarbeit wie in Open Source-Projekten, aber auch Erwerbsarbeit untersucht werden können.

### **2.3.1. Arbeitsmarktfunktionen und implizit angenommener Arbeitsprozess**

Ausgehend von der Forschungsfrage, wie die virtuelle Arbeit erfolgt, wird zuerst auf Ansätze aus der Arbeitsmarktforschung zurückgegriffen. Aus einer funktionalen Sicht wird üblicherweise der Arbeitsmarkt aus einer ökonomischen Perspektive über Arbeitsmarktfunktionen untersucht. Ausgangspunkt ist dabei die Transformation von Arbeitskraft in Arbeitsleistung in einer Organisation, die einen implizit angenommenen Arbeitsprozess darstellt. Die Transformation von Arbeitskraft in Arbeitsleistung wurde bisher im Kontext der Erwerbsarbeit gesehen (s.o.). Es wurde dabei der Frage nachgegangen, wie ein Arbeitgeber Arbeitnehmer für Aufgaben erhält und Arbeitnehmer die Arbeitskraft in Arbeitsleistung überführt oder überführen kann (vgl. Becher et al. 2005). Aus dieser Perspektive der Erwerbsarbeit, aufgrund von arbeitsvertragstheoretischen Überlegungen und des Fokus' auf Unternehmen lassen sich bei der Transformation Arbeitsmarktfunktionen wie „Allokation“, „Leistungserstellung“, „Kontrolle“ und „Gratifikation“ analytisch ableiten und inhaltlich aufzählen (vgl. Becher et al. 2005: 28ff.):

Die *Allokationsfunktion* bezeichnet einerseits eine organisationsexterne Allokation von Arbeitnehmern auf Stellen, d.h. von organisationsexternen Arbeitskräften zu einer Organisation und zu einem Arbeitsbereich. Damit ist eine Rekrutierung mit Anwerbung und Einstellungsverhandlungen sowie den konkreten Eintritt in ein Unternehmen oder eine Organisation verbunden, der normalerweise mit einer Zuweisung von groben Arbeitsbereichen einhergeht. Dabei ist der Eintritt die Voraussetzung, dass in einer Organisation/einem Unternehmen überhaupt gearbeitet wird. Andererseits tritt neben die organisationsexterne eine organisationsinterne Allokation der Arbeitsaufgaben, aber auch Aufgabenbereiche auf die Organisationsmitglieder bzw. jeweilige Arbeitskraft. Die konkreten Arbeitsaufgaben sind mit dem Arbeitsprozess verbunden und ändern sich im Laufe der Organisationszugehörigkeit oftmals. Diese interne Allokation kann mit einer Arbeitsverteilung gleichgesetzt werden. Darunter kann auch die Arbeitsverteilung durch Postenbesetzung verstanden werden, d.h. die Zuordnung von Posten mit bestimmten Aufgaben- und Verantwortungsbereichen zu Arbeitskräften (vgl. Becher et al. 2005). Der Austritt aus einer Organisationsform wird in dieser Untersuchung vernachlässigt, da dieser keine direkten Konsequenzen für den Arbeitsprozess hat.

Die *Leistungserstellung* bezieht sich auf die konkrete Transformation von Arbeitskraft in Arbeitsleistung, d.h. die konkrete Erstellung eines Arbeitspakets durch einen Arbeitnehmer. Die Leistungserstellung ist also der konkrete Prozess der Umwandlung des Arbeitskraftpotenzials einer Arbeitskraft über Arbeitsanstrengung in eine individuelle Arbeitsleistung bzw. ein Arbeitspaket, das einen Anteil des späteren Gesamtprodukts darstellt.

Die *Kontrolle* und die *Gratifikation* beziehen sich auf die Frage, wie eine Organisation/ein Unternehmen ein hochwertiges Arbeitsergebnis von den Arbeitnehmern erhalten kann. Die Kontrolle bezieht sich dabei auf die Qualitätsfeststellung des Arbeitsergebnisses. Gratifikation bezieht sich auf Anreizsysteme, die die Motivation der Arbeitnehmer erhöhen sollen.

Damit beschreiben die Arbeitsmarktfunktionen im Prinzip einen Prozess von der Suche eines potenziellen Arbeitnehmers über den Eintritt in ein Unternehmen mit Stellenübernahme bis hin zur eigentlichen Erwerbstätigkeit. Bei externen Arbeitsmärkten steht die Allokation von Arbeitnehmern auf Stellen in Unternehmen im Vordergrund, bei internen Arbeitsmärkten und Arbeitsnetzen dagegen alle vier Funktionen (vgl. Becher et al. 2005, 2007).

Die vier Funktionen bilden dabei größtenteils die Perspektive des Managements bzw. der Personalabteilung eines Unternehmens ab. Dahinter steht die Frage des Managements bzw. des Unternehmens, wie Rekrutierung/Eintritt, Leistungserstellung, Kontrolle und Gratifikation ablaufen sollen, so dass ein effektiver und u.U. effizienter Arbeitsprozess der Transformation von Arbeitskraft in Arbeitsleistung erfolgen kann. Die Arbeitsmarktfunktionen können aber auch auf andere Arbeitsarten, wie Arbeitsnetze bzw. Open Source-Projekte, als Betrachtungsfolie verwendet werden (vgl. Becher et al. 2005).

Gerade die genannte Erwerbsarbeitsperspektive des Managements bzw. des Unternehmens beschreibt die Arbeitszusammenführung zu einer Gesamtarbeitsleistung, wobei diese Perspektive automatisch bei der Gratifikations- und Kontrollfunktion eingenommen wird. Die konkrete arbeitsteilige Zusammenarbeit wird nicht direkt betrachtet, sondern die Ausrichtung der Arbeitnehmer auf ein Ziel. Die Gratifikationsfunktion soll die Motivation und damit die Arbeitsleistung und Qualität von Arbeitspaketen erhöhen, die dann in der Kontrollfunktion kontrolliert und zu einer Gesamtarbeitsleistung zusammengefügt werden. Daher sind Gratifikation und Kontrolle zwei Seiten einer Medaille und mit der Management- bzw. Unternehmensperspektive direkt verbunden. Diese beiden Funktionen können in dieser Form nicht, die Arbeitszusammenführung als dahinterliegende Idee dagegen sehr wohl in den Analyserahmen übernommen werden. Den Abschluss bildet demnach die Zusammenführung der erstellten Arbeitspakete zu einer organisationalen, abgeschlossenen Arbeitsleistung, die von den Arbeitspaketen geringen Umfangs der Mitarbeiter zu unterscheiden ist.

Zusammengefasst sind als Funktionen des impliziten Arbeitsprozesses, die auch für Nicht-Erwerbsarbeit verwendet werden können, folgende Funktionen zu nennen, die in einer nicht unbedingt zeitlichen, aber logischen Reihenfolge zueinander stehen:

- Eintritt/Rekrutierung bzw. externe Allokation (Aufnahme der Arbeitenden in eine Organisation)
- Arbeitsverteilung bzw. interne Allokation (Verteilung von konkreten Arbeitspaketen auf die Arbeitenden)
- Leistungserstellung (Bearbeitung von Arbeitspaketen)
- Arbeitszusammenführung (Zusammenführung von einzelnen Arbeitspaketen zu einer organisationalen Arbeitsleistung)

### **2.3.2. Verschiedene Phasenmodelle der Softwareentwicklung**

Für eine detailliertere Teilprozesserstellung werden Ansätze aus der Softwareentwicklung herangezogen. Aus einer zeitlichen Sicht der Phasenmodelle wird weitestgehend ein idealer Arbeitsprozess projiziert. Da der Ablauf der Softwareentwicklung bei den virtuellen Arbeitsnetzen eine Rolle spielt, werden die Phasenmodelle der Softwareentwicklung mit in die Betrachtung einbezogen. Damit kann die obige schematische Betrachtung anhand dieser Modelle detaillierter ausgestaltet werden, um später für die Untersuchung des Open Source-Projekts verwendet zu werden. Eine Annäherung an die Softwareentwicklungsprozesse erfolgt über Phasenmodelle der Softwareentwicklung, die als Vorstellungen bzw. abstrakte, ideale Schemata den Ablauf einer Softwareentwicklung wiedergeben (vgl. Sommerville 2001; Balzert 1996).

Drei Softwareentwicklungsmodelle werden nach ihrer formalen, vorher geplanten Strukturiertheit, dem Grad der Involvierung der Nutzer/Kunden sowie der Parallelität der ablaufenden Einzeltätigkeiten unterteilt und als ideale Vorstellungen beschrieben. Eins der drei Modelle beschreibt entweder größtenteils oder alle teilweise den Arbeitsprozess in einem virtuellen Arbeitsnetz mit Softwareentwicklung. Der Nutzereinbezug in den Arbeitsprozess

bestimmt die Softwareproduktion, wie oben argumentiert wurde (s. Kap. 2.2.3.). Da der Nutzereinbezug in den verschiedenen Softwareentwicklungsmodellen unterschiedlich ist, wird dieser im Folgenden auch dargestellt.

Die drei Softwareentwicklungsmodelle sind erstens das in der industriellen Softwareentwicklung verwendete Kaskaden- bzw. Wasserfallmodell, das am stärksten formal strukturiert ist. Zweitens besteht ein Bereich mit verschiedenen Softwareentwicklungsmodellen, die sich nach Involviertheit der Nutzer/Kunden und der Parallelität der Softwareentwicklung unterscheiden. Diese verschiedenen Modelle können unter dem Überbegriff evolutionäre Entwicklungsmodell zusammengefasst werden. Zentrale Merkmale dieser Modelle werden herausgearbeitet, da angenommen werden kann, dass das Vorgehen in virtuellen Arbeitsnetzen u.a. diesen evolutionären Modellen ähnelt. Dabei stehen diese evolutionären Modelle zwischen einer formalen Strukturierung im Wasserfallmodell und drittens einem Open Source-Softwareentwicklungsmodell. Dieses Open Source-Softwareentwicklungsmodell ist am wenigsten formal strukturiert und relativ ergebnisoffen. Es wird an dieser Stelle dargestellt, da ein Open Source-Projekt untersucht wird. Es kann daher wegen fehlender differenzierter Phasenaufteilung (im Prinzip bestehen nur zwei Phasen) und seiner generischen Entstehung aus empirischem Material nicht direkt für eine idealisierten Arbeitsprozess verwendet werden (Boehm 2002: 65; Abrahamsson, et al. 2002: 16; Sommerville 2001: 9, 44ff.; Balzert 1996; Feller, Fitzgerald 2002). Nur das Wasserfallmodell und bedingt das evolutionäre Entwicklungsmodell werden bei einer ersten Verfeinerung des impliziten Arbeitsprozesses der Arbeitsmarktfunktionen herangezogen, soweit Elemente der Modelle einfließen können. Anhand des Open Source-Softwareentwicklungsmodell wird für eine spätere empirische Betrachtung überprüft, ob dieses Modell mit dem erstellten, idealisierten Arbeitsprozess als Analyserahmen zusammengeht und dieses wird ggf. nochmals verfeinert.

### 2.3.2.1. *Kaskaden- bzw. Wasserfallmodell*

Ausgangspunkt ist das Kaskaden- bzw. Wasserfallmodell der stark strukturierten Softwareentwicklung, das üblicherweise in industriellen Großprojekten angewendet wird. Es stellt ein Beispiel des ersten formalisierten Softwareentwicklungsmodells dar, das historisch als Reaktion auf die „Software-Krise“ der 1960er und 1970er Jahre entstand. In dieser Krise wurde festgestellt, dass die bestehenden informalen Softwareentwicklungsstrategien bei komplexen Programmen ineffizient sind, zu mangelnder Qualität und hohen Kosten führen und die Softwareentwicklung nicht mit der Industrialisierung der Hardwareproduktion Schritt hält (Sommerville 2001: 4; vgl. Friedrich 1992). Dieses Softwareentwicklungsmodell ist durch eine strukturierte, abgeschlossene Phasenabfolge gekennzeichnet, die von vornherein den Ablauf der Softwareentwicklung festlegen soll (s. Abb. 2-1). Ziel des Phasenmodells ist es, Kosten, Termineinhaltung, Qualität und Anforderungsübereinstimmung kontrollieren zu können (Balzert 1996: 25ff.; vgl. Sommerville 2001). Damit ist eine starke Strukturierung, mangelnde Rückbezüge auf frühere Phasen und eine hierarchische Planungsstrategie (Top-down) verbunden und somit eher in Großunternehmen vorzufinden (vgl. Welz, Ortman 1992).

In der Planungsphase werden erste Kosten-Nutzen-Berechnungen nach Vorstudien und wünschbaren Softwarefunktionsumfängen in einem Lastenheft erstellt, um entscheiden zu können, ob ein Softwareentwicklungsprojekt durchgeführt werden soll (s. Abb. 2-1). Falls das Projekt bejaht wird, wird in der Zieldefinitionsphase eine genaue Anforderungsdefinition und –analyse der zu erstellenden Software durchgeführt und in einem Pflichtenheft verfeinert. Bei der Erstellung spezifisch angepasster Individualsoftware für ein Unternehmen/einen Betrieb/eine Abteilung werden üblicherweise in dieser Phase die Nutzer/Kunden mit einbezogen, bei Standardsoftware ist die Marketingabteilung das Sprach-

rohr des Nutzers. Die Entwurfsphase ist durch eine abstrakte Beschreibung der Systemkomponenten und der Schnittstellen zwischen verschiedenen Systemkomponenten (Softwarearchitektur) sowie der Datenstruktur der Software bestimmt. Das Programmieren der Vorgaben aus der Entwurfsphase in einer Programmiersprache steht in der Implementierungsphase im Vordergrund, was auch als eine Art Operationalisierung bezeichnet werden kann. In der Abnahme- und Einführungsphase (Testphase) wird das fertige Gesamtprodukt dem Auftraggeber übergeben und von diesem auf Fehler und Übereinstimmung mit den Anforderungen der Definitionsphase getestet und überprüft. Anschließend erfolgt die Wartungs- und Pflegephase, bei der die Software ggf. korrigiert, verbessert oder an neue Gegebenheiten angepasst wird.

**Abb. 2-1:** Phasenabfolge des Wasserfallmodells; nach Balzert 1996, leicht verändert

Planungsphase

⇒ Zieldefinitionsphase (Anforderungsdefinition und -analyse)

⇒ Entwurfsphase

⇒ Implementierungsphase

⇒ Abnahme- und Einführungsphase (Testphase)

⇒ Wartungs- und Pflegephase

Das traditionelle Wasserfallmodell wurde ohne Rücksprünge konzipiert. Bei einem angepassten Modell sind Rücksprünge und Veränderungen nach Abschluss der einzelnen Phasen möglich, erfolgen aber oft nur in die vorige Phase (vgl. Balzert 1996; Sommerville 2001).

Mehrere, z.T. bis zu hundert, Personen arbeiten beim Wasserfallmodell in der industriellen Softwareentwicklung über ein bis mehrere Jahre an der Software. Dagegen gibt es kleine Softwareentwicklungsprojekte mit wenigen Beteiligten und recht kurzer, teilweise nur einen Monat währender Laufzeit (vgl. Reifer 2000; Abrahamsson, et al. 2002). Diese kleinen Projektteams folgen eher anderen Softwareentwicklungsmodellen, die im Folgenden dargestellt werden.

#### 2.3.2.2. *Evolutionäres Softwareentwicklungsmodell bzw. Spiralmodell*

Die Unterteilung der Softwareentwicklung in Phasen ist zwar bei bekannten Aufgabenstellungen möglich. Beim Betreten von „Neuland“ ist das Wasserfallmodell jedoch eher ungeeignet, da die Anpassung der Software an geänderte Anforderungen nur nach Beendigung einer Phase möglich ist, und Phasen können mehrere Monate umfassen.

Als Reaktion auf die Unzulänglichkeiten des Wasserfallmodells sind mittlerweile seit Ende der 1990er Jahre neuere Konzepte der Softwareentwicklung (z.B. Rapid Prototyping, Spiral Development, Reuse-Oriented Development) entstanden, die eine Auflockerung und Weiterentwicklung des recht starr strukturierten Wasserfallmodells darstellen. Die Phasen der Planung/Zieldefinition und der Abnahme/Einführung (nach Erstellung einer Gesamtarbeitsleistung) stellen den groben Rahmen des inkrementellen Entwicklungsprozesses dar und werden in den nachgenannten Modellen „evolutionär“, d.h. in einem kreativen Freiraum, entwickelt (Bsp. Evolutionary Approach: Specification, Evolutionary Development, System Testing). Vielmehr nimmt der Bereich der konkreten, parallelen Entwicklung (Evolutionary Development) einen größeren kreativen Raum ein, wobei eine starre, detaillierte Planung nicht stattfindet. Zum einen werden die Nutzer bzw. Auftraggeber stärker in die

Softwareentwicklung eingebunden, zum anderen sind ständige Rekursionen bzw. Feedback-Schleifen eingebaut, die ein ständiges Anpassen an neue Gegebenheiten erlauben. Eine solche Gegebenheit ist z.B. die Ad hoc-Wiederverwendung und Anpassung von schon vorhandenen Softwarekomponenten. Bei der evolutionären Softwareentwicklung wird z.B. von einer geplanten groben Spezifikation ausgehend ein Prototyp entwickelt, wiederholt überprüft, bewertet und weiter verändert.

Beim Spiralmodell, um ein zentrales Beispiel dieses Modells herauszugreifen, werden vier Phasen während des Projektablaufs nacheinander wiederholt und in kurzer Zeit durchlaufen (s. Abb. 2-2). Zuerst werden allgemeine Anforderungen, Alternativen und Einschränkungen identifiziert, als zweite Phase Alternativen bewertet/Risiken identifiziert, dann Software entwickelt und zum Schluss die nächste Phase geplant und wieder zur ersten Phase gesprungen. Danach wiederholt sich die Spirale von vorn, wobei die Softwareentwicklung bei jedem Durchlaufen der Phasen immer detaillierter, das Softwareprodukt immer fehlerfreier wird und sich stärker an den allgemeinen Anforderungen orientiert. Das Spiralmodell beginnt wie beim Wasserfallmodell mit einem Auftrag mit ersten Zieldefinitionen und endet mit der Abnahme der Endversion durch einen Auftraggeber (Sommerville 2001: 9, 44ff.; Anderson, et al. 1998; Beck 2000; Abrahamsson, et al. 2002).<sup>16</sup>

**Abb. 2-2:** Entwicklungsphasen in einem Spiralmodell, leicht verändert nach Sommerville 2001

Start

- ⇒1. Anforderungen, Alternativen, Einschränkungen identifizieren
- ⇒2. Alternativen bewerten, Risiken identifizieren
- ⇒3. Entwicklung
- ⇒4. Planung der nächsten Phase
- ⇒Ende nach mehrmaligen Durchlaufen der obigen 4 Phasen

Die verschiedenen evolutionären Softwareentwicklungsmodelle unterscheiden sich nach den vorgeschriebenen weiteren Phasen, d.h. nach dem Ausmaß der Offenheit bzw. Strukturiertheit. Einige Projekte bestehen überwiegend aus einer relativ ergebnisoffenen Spiralentwicklung, andere haben noch einen starken Planungs- und Testanteil. In Unternehmen laufen die Softwareentwicklungsprojekte im Allgemeinen für eine bestimmte Zeit, weswegen eine grobe Planungs- und abschließende Abnahmephase normalerweise vorhanden ist (Sommerville 2001: 9, 44ff.; Anderson, et al. 1998; Beck 2000; Abrahamsson, et al. 2002).

### 2.3.2.3. Softwareentwicklung im Open Source-Projekt

Die Softwareentwicklung im Open Source-Projekt als drittes Softwareentwicklungsmodell verläuft vollkommen konträr zum Wasserfallmodell. Die Softwareentwicklung läuft vollständig parallel in dezentralisierten kleinen Gruppen ab (s. Abb. 2-3), was durch die Modularität der Softwarearchitektur weitestgehend ermöglicht wird (vgl. Abrahamsson et al. 2002; Jörgensen 2001; Mockus et al. 2002; Narduzzo, Rossi 2003; German 2003). Außer-

---

<sup>16</sup> Ähnliche Anwendungen des Spiralmodells sind z.B. Extreme Programming oder Agile Development. Zwei Programmierer schreiben bei wöchentlichen Kontrolltreffen mit den Nutzern Testsoftware nach den Vorgaben und entwickeln dann die eigentliche Software, die den Anforderungen der Testsoftware genügt.

dem besteht eine starke Verknüpfung von Entwicklern und Nutzern, wobei die Entwickler oft auch Nutzer sind (s. Kap. 5.4.). Dieses Phasenmodell wurde nicht wie die beiden oben genannten Modelle von vornherein geplant, sondern hat sich im Laufe des Bestehens der Open Source-Projekte neben den anderen in einem eigenen Bereich entwickelt. Daher wird hier auf empirische Untersuchungen zurückgegriffen.

Das Vorgehen unterscheidet sich vom Spiralmodell nur in bestimmten Punkten. Der Unterschied liegt in einem vollständigen Versuchs- und Irrtumsprozess und in der Anwendung einer Feedback-Schleife auf einer übergreifenden, organisationalen Ebene. Die übergeordnete Planung von Produktionsrelease und darüber hinausgehenden, weiteren Schritten erfolgt idealerweise parallel zur normalen Softwareentwicklung. Es wird von einer kontinuierlichen Softwareentwicklung ohne definierten Anfang und definiertes Ende ausgegangen, wobei immer wieder Zwischenergebnisse als stabile und lauffähige Programme publiziert werden. Ein reines Spiralmodell dagegen hat einen definierten Anfang und ein definiertes Ende (vgl. Feller, Fitzgerald 2000, 2002; Yamauchi et al. 2000; Jörgensen 2001).

Diese parallele Bearbeitung in Open Source-Projekten auch von verschiedenen Entwicklungsversionen kann in einer Abstraktion in zwei sich abwechselnde Entwicklungsphasen nach Jörgensen (2001) und Mockus et al. (2002) unterteilt werden, die zu einem mehr oder minder stabilen Zustand führen (s. Abb. 2-3). Auf der einen Seite erfolgt eine ständig in Veränderung befindliche, dynamische Softwareentwicklung, die zu einem Entwicklungsrelease führt (vgl. Jörgensen 2001: 328). Der Entwicklungsrelease steht mit freien und kreativen Quellcodeveränderungen im Versionsmanagementsystem in Verbindung, wobei keine stark begrenzenden Einschränkungen wirken. Die Softwareerstellung ist durch einen iterativen/rekursiven, unstrukturierten Prozess der Programmierung bestimmt, d.h. dieser erfolgt Schritt für Schritt mit vielen kleinen Veränderungen in einem Versuchs- und Irrtumsprozess. Die Mitarbeiter haben ständig neue Aufgaben durch auftretende Fehler oder Probleme zu bearbeiten und zu lösen, wobei sie versuchen, ein selbstgesetztes Arbeitsziel zu erreichen. Der Projektbeteiligte arbeitet normalerweise zu Hause an seiner Software an einer lokalen Kopie aus dem Versionsmanagementsystem. Der geschriebene Quellcode wird vom Programmierer zu Hause auf seine Funktionsfähigkeit getestet, d.h. es erfolgt normalerweise eine Selbstkontrolle (s. Jörgensen 2001 für das FreeBSD-Projekt; Mockus et al. 2002 für Apache und Mozilla; German 2003 für das Gnome-Projekt). Danach wird der Quellcode in das Versionsmanagementsystem als momentaner Entwicklungsrelease gestellt und ein neues Arbeitspaket übernommen.<sup>17</sup>

Auf der anderen Seite wird in einer zweiten, stark abgegrenzten Phase aus dem Entwicklungsrelease eine abgeschlossene, statische, fehlerfreie, offizielle Endversion (Production Release) erstellt, die zu einem spezifischen Termin als stabile Version des gemeinsamen Softwareprodukts in lauffähiger Maschinensprache mit Dokumentationen und Übersetzungen veröffentlicht wird. In der Erstellungsphase der offiziellen Endversion werden nur Fehler aufgedeckt und beseitigt (Parallel Debugging), wobei entweder ein Releasekoordinator, eine Kommission oder alle Projektbeteiligten das Vorgehen planen und managen. Die Softwareerstellung wird in dieser Phase durch einen Releaseplan als soziale Norm bestimmt, der eine gestufte Beschränkung der Veränderungsmöglichkeiten in

---

<sup>17</sup> Es besteht ein großer Unterschied zwischen der Softwareerstellung im Open Source-Projekt und der traditionellen, stark strukturierten Softwareentwicklung in Unternehmen. Ist der Ablauf der Softwareentwicklung im Unternehmen durch Planung, Analyse, Design, Implementation bestimmt, so kann die Softwareerstellung im Open Source-Projekt in Programmierung, Review, Pre-Commit Test, Development Release, Parallel Debugging und Production Release aufgliedert werden (Jörgensen 2001; Feller, Fitzgerald 2002: 101ff.).

dem Versionsmanagementsystem darstellt.<sup>18</sup> Die Arbeit im Versionsmanagementsystem wird nicht technisch eingeschränkt. Sobald eine hohe Fehlerfreiheit garantiert werden kann, wird der Release von speziellen Softwarepaketierern<sup>19</sup> zusammengebaut und für die jeweiligen Betriebssysteme in Maschinensprache übersetzt (vgl. Jörgensen 2001: 328; Tuomi 2001; German 2004; Erenkrantz 2003).

**Abb. 2-3:** Ablauf eines Open Source-Projekts; leicht verändert nach Jörgensen 2001; Mockus 2002



#### 2.3.2.4. *Entwicklungsphasen der Softwareentwicklungsmodelle und Arbeitsmarktfunktionen*

Die Softwareentwicklungsmodelle als analytische, ideale Schemata beschreiben im Prinzip die Transformation von Arbeitskraft in Arbeitsleistung als Arbeitsprozess bei der Softwareentwicklung. Das Wasserfallmodell und das evolutionäre Modell stehen im Mittelpunkt dieser Phasenmodellbetrachtung, da sie zeitlich aufeinanderfolgende Phasen bezeichnen. Dabei ähneln diese beiden Vorgehensmodelle der Softwareentwicklung als ideale Vorstellungen dem ingenieurwissenschaftlichen Konstruieren. Das Wasserfallmodell und seine Abarten, aber auch die evolutionären Ansätze sind als analytische Zweck-Mittel-Schemata zu sehen, wobei das Vorgehen in Phasen eingeteilt wird, die als eine Art Checkliste das Arbeiten anleiten. Davon unabhängig ist das reale, kreative Entwicklungsvorgehen zu betrachten (Glock 1997: 18ff.; 35f.).

Diese zeitlichen Phasenabfolgen werden beim Wasserfallmodell durch Rücksprünge und beim evolutionären Modell durch das wiederholte Durchlaufen von allgemein gehaltenen Arbeitsschritten aufgelöst. Auch Welz und Ortmann (1992: 81f.) haben festgestellt, dass sich zwar Unternehmen an Phasenmodellen orientieren, diese Modelle aber immer wieder situativ angepasst werden. Eine stringente und strikte Phasenabfolge entspricht damit nicht dem realen Entwicklungsvorgehen. Gerade das Open Source-Entwicklungsmodell gibt das reale, kreative Entwicklungsvorgehen wieder, da es aus dem empirischen Feld generiert wurde. Obwohl beim Open Source-Entwicklungsmodell die Parallelität der Bearbeitung zentral ist, wurde das Vorgehen mittels eines Phasenmodells erfasst. Eine parallele Bearbeitung kann auch bei einer Softwareentwicklung nach dem Wasserfallmodell oder dem evolutionären Modell vorkommen. Das Wasserfallmodell oder das evolutionäre Modell deswegen zu verwerfen, ergibt keinen Sinn, da in beiden Modellen prinzipielle, aufeinander logisch aufbauende Arbeitsschritte benannt werden. Diese Arbeitsschritte können für den Analyserahmen eines idealisierten, analytischen Arbeitsprozesses verwendet werden. Im Weiteren wird auf das Wasserfallmodell und das evolutionäre Modell eingegan-

---

<sup>18</sup> Es gibt verschiedene Einzelphasen, in denen nacheinander die Anzahl der Quellcodeveränderungen bis zu einem kompletten Einfrieren des Dateiablagesystems verringert werden.

<sup>19</sup> Die Paketierer sind Personen mit schnellen Rechnern, die meistens bei Distributoren angestellt sind.

gen, um diese Arbeitsschritte für eine Übernahme in den Analyserahmen herauszupräparieren. Da ein Open Source-Projekt mittels dieses Analyserahmens untersucht wird, wird der Analyserahmen anhand des Open Source-Entwicklungsmodells nicht in diesem, sondern im folgenden Unterkapitel auf seine Verwendungsmöglichkeit überprüft.

Eine große Rolle spielt in den Softwareentwicklungsmodellen, wie oben angedeutet wurde, die Involvierung der Nutzer/Kunden. Beim Wasserfallmodell spielen Endnutzer/Kunden nur am Anfang und Ende des Arbeitsprozesses als Auftragserteilung und –endkontrolle eine große, bei der konkreten Softwareentwicklung jedoch kaum eine Rolle (s.o.). Das evolutionäre Softwareentwicklungskonzept versucht verstärkt, Nutzer/Kunden in die Identifikation des Anfangsproblems einzubeziehen, indem Änderungen und wiederholte Rückbezüge auf vorherige Entscheidungen und Arbeitspakete sowie allmähliche Verfeinerungen während des Entwicklungsprozesses anhand von Zwischenprodukten angestrebt werden (Glock 1997: 18f.).

Generell können in den beiden Modellen die drei Phasen (1) „Planung“, (2) „Quellcodeprogrammierung“ und (3) „Abnahme der Endversion durch einen Auftraggeber nach der Erstellung einer Gesamtarbeitsleistung“ herauskristallisiert werden. Die Phasen der Softwareentwicklungsmodelle mit ihren Tätigkeitsschwerpunkten ähneln den oben genannten Arbeitsmarktfunktionen und können diesen zugeordnet werden. Die Arbeitsmarktfunktion Eintritt/Rekrutierung (externe Allokation) wird aber von den Softwareentwicklungsmodellen nicht beschrieben.

Bei der Planung kann erstens eine allgemeine, übergreifende Anforderungserstellung am Anfang von einer späteren, detaillierten Planung nahe der konkreten Quellcodeerstellung unterschieden werden. Beim Wasserfallmodell sind die Aktivitäten bei der Planungs- und Zieldefinitionsphase mit Anforderungsdefinition und -analyse ähnlich, gehen z.T. ineinander über und beschreiben die anfängliche Anforderungserstellung (vgl. Balzert 1996; Glock 1997: 18ff.). In einer Entwurfsphase wird dann das Vorgehen detailliert beschrieben. Ähnlich ist es im evolutionären Modell, bei dem am Anfang ein Auftrag mit allgemeinen Anforderungen als Erstellungsziel steht, der dann mittels Alternativenidentifizierung und -bewertung detaillierter geplant wird. Implizite Bezüge bestehen zwischen Planungsphase und der Arbeitsmarktfunktion Arbeitsverteilung (interne Allokation), wobei eine Arbeitsverteilung eine vorherige Planung impliziert. Umgekehrt folgt nach einer Planung normalerweise eine Verteilung der prospektiven Arbeitspakete auf die Arbeitenden. Folglich ist die Arbeitsverteilung als Teilprozess des Analyserahmens in die Planung mit einzubeziehen. Ein Kunden-/Endnutzerbezug ist bei der Planungs-, Zieldefinition und bedingt bei einer detaillierten Planung vorhanden.

Zweitens nennt sich die Phase Quellcodeprogrammierung als konkrete Softwareentwicklung im Wasserfallmodell Implementierungsphase, im Spiralmodell Entwicklung. Sie ähnelt stark der Leistungserstellungsfunktion. Sieht das Wasserfallmodell keinen Kundeneinfluss, ist dies beim evolutionären Modell u.a. möglich.

Drittens besteht beim Wasserfallmodell und dem evolutionären Modell eine Abnahmephase, in der das Softwareprodukt als Gesamtarbeitsleistung erstellt und vom Auftraggeber/Kunden/Endnutzer übernommen wird. Diese Phase ist der Funktion Arbeitszusammenführung der einzelnen Arbeitspakete zu einem lauffähigen Arbeitsprodukt sehr ähnlich.

### 2.3.3. Teilprozesse des Softwareentwicklungsprozesses als Analyserahmen

Nachdem die Arbeitsmarktfunktionen und die Phasenmodelle der Softwareentwicklung vorgestellt wurden, erfolgt in diesem Unterkapitel eine Zusammenführung von Funktionen und Phasen zu einem idealisierten Arbeitsprozess (Workflow) der Softwareentwicklung (s. Tab. 2-1). Dieser idealisierte Arbeitsprozess wurde durch die Verwendung dieser verschiedenen Ansätze detailliert zusammengestellt. Dieser idealisierte Arbeitsprozess wird dabei aus Idealvorstellungen über Arbeitsprozesse entwickelt. Funktionen und Phasen sollen dabei Teilprozesse der Softwareentwicklung benennen, die später empirisch untersucht werden können. Die Teilprozesse bilden den Arbeitsprozess schematisch ab. Diese Teilprozesse befinden sich dabei in einer logischen Reihenfolge, sind aber nicht als zeitlich-aufeinanderfolgende Phasen aufzufassen. Dabei werden in den Teilprozessen die Kerntätigkeiten betrachtet, die dem Ziel dieses Teilprozesses zuzuordnen sind. Teilprozesse können zeitlich parallel zueinander ablaufen, da Funktionen ohne Zeitbezug eine Rolle spielen. Diese Funktionen stehen dabei über einem implizit angenommenen Arbeitsprozess in einer logischen Reihenfolge. Außerdem verhindert die zeitliche Phasenabfolge im Wasserfallmodell und im evolutionären Modell eine adäquate Betrachtung der Softwareentwicklung im empirischen Feld. Die genannten logischen Arbeitsschritte ohne zeitlichen Bezug lassen sich aber verwenden. Im evolutionären, aber auch im Open Source-Entwicklungsmodell als Hauptuntersuchungsgegenstand ist paralleles Arbeiten in verschiedenen Phasen möglich. Daher ist eine Trennung zwischen den Teilprozessen nicht immer eindeutig möglich, weswegen die Teilprozesse als analytisch-logische Einteilungen zu verstehen sind.

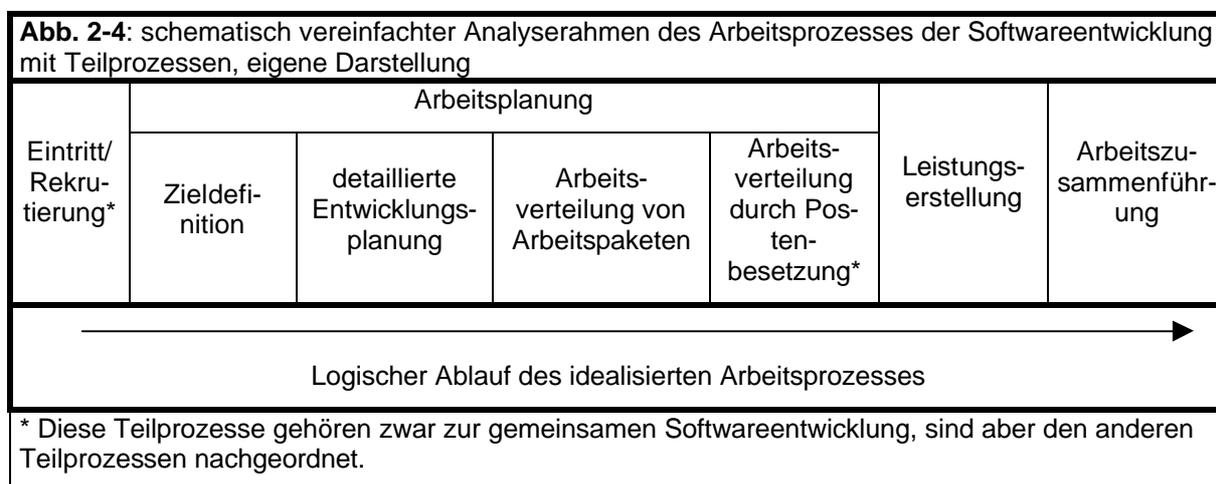
**Tab. 2-1:** Anpassungsschritte zum Erhalt eines idealisierten Arbeitsprozesses mit Teilprozessen, eigene Darstellung

Anpassungsschritte	Funktionen und Phasen als Teilprozesse			
<b>1.</b> Arbeitsmarktfunktionen	Allokation		Leistungserstellung	Kontrolle Gratifikation
<b>2.</b> impliziter Arbeitsprozess	Externe Allokation	Interne Allokation (Arbeitsverteilung)	Leistungserstellung	Arbeitszusammenführung
<b>3a.</b> Anpassung an Softwarephasenmodelle	Eintritt/Rekrutierung	Planung und Arbeitsverteilung*	Leistungserstellung (Softwareprogrammierung)	Arbeitszusammenführung
<b>3b.</b> Ausdifferenzierung nach Phasenmodellen	Eintritt/Rekrutierung	Unterprozesse: Zieldefinition, detaillierte Entwicklungsplanung, Arbeitsverteilung von Arbeitspaketen und Posten	Leistungserstellung (Softwareprogrammierung)	Arbeitszusammenführung

\* Da Arbeitsverteilung ohne Planung nicht möglich

Im Folgenden wird der Analyserahmen des idealisierten, analytischen, logisch abfolgenden Arbeitsprozesses mit seinen Teilprozessen vorgestellt (s. Abb. 2-4). Da ein Open Source-Projekt untersucht wird (s. Kap. 5) und das Open Source-Entwicklungsmodell aus dem empirischen Feld generiert wurde, wird zusätzlich die Verwendungsmöglichkeit des Analyserahmens überprüft, d.h. wie das Open Source-Entwicklungsmodell mit den vorher entwickelten Teilprozessen des Arbeitsprozesses zusammengehen. Es deuten sich einige ähnliche Elemente im Open Source-Entwicklungsmodell und den beiden anderen Modellen an, so dass das Open Source-Entwicklungsmodell weitestgehend in den Analyserahmen eingepasst werden kann. Die Teilprozesse stehen in einer logischen Reihenfolge eines idealisierten, analytischen Ablaufs eines Arbeitsprozesses, was die Integration der parallelen Entwicklung in einem Open Source-Projekt in den Analyserahmen erleichtert.

Da dieser idealisierte Arbeitsprozess aus verschiedenen Ansätzen mit unterschiedlichem Detaillierungsgrad entwickelt wurde, sind nicht alle Teilprozesse auf der gleichen Detaillierungsebene, was im Folgenden dargestellt wird. Gerade der Teilprozess „Arbeitsverteilung durch Postenbesetzung“ und der Teilprozess „Eintritt/Rekrutierung“ gehören nicht auf



- Der Teilprozess *Eintritt/Rekrutierung* ist der externen Allokationsfunktion nachempfunden. Obwohl dieser Teilprozess im Wasserfall- und evolutionären Modell, aber auch im Open Source-Entwicklungsmodell nicht erwähnt wird, ist dieser Teilprozess auch wichtig für den Arbeitsprozess. Denn ohne (virtuell) zusammenkommende Arbeitende z.B. in einer Organisation ergibt sich kein gemeinsames Arbeitsprodukt. Dieser Teilprozess ist dabei aber eine wichtige und notwendige, aber nicht hinreichende Voraussetzung für diesen Arbeitsprozess. Ohne diesen Teilprozess könnten die nachfolgenden Teilprozesse nicht ablaufen, doch wird in diesem Teilprozess kein direkter Einfluss auf die Softwareentwicklung genommen. Dies zeigt sich, indem nicht alle Softwareentwickler, sondern nur Projektleiter und potenzielle Softwareentwickler in diesem Teilprozess involviert sind. Zum einen wird unter diesem Teilprozess der Ablauf vom ersten Kontakt bis zum Eintritt im Sinne eines Mitgliedschaftsbeitritts in eine bestehende Zusammenarbeit von mehreren Akteuren verstanden. Diese Akteure können dabei u.U. eine Organisationsform bilden, wobei eine Mitgliedschaft in dieser Organisation errungen wird. Zum anderen kann in diesem Teilprozess eine Mitgliedschaft qua Gründung bzw. Konstituierung einer Zusammenarbeit in einer Organisation zugewiesen werden. Die Mitgliedschaft kann später, nach Fertigstellung eines Arbeitspakets, wobei der Arbeitsprozess weiterlaufen kann, aber auch des Endprodukts aufgehoben werden. Darunter fallen z.B. Eintritte in Open Source-Projekte (s. Kap. 5). Damit ist als Ziel eine Vollintegration bzw. Vollmitgliedschaft in der Organisation verbun-

den. Wie aus obigen Kapiteln bekannt ist, arbeiten auch Endnutzer wenigstens teilweise am Arbeitsprozess z.B. in der Softwareentwicklung mit. Vollintegrierte Mitarbeiter sind daher Personen, die sich zu der Organisation zugehörig fühlen und von den anderen Mitgliedern als Mitglied anerkannt werden. Die Anerkennung als Mitglied erfolgt weitestgehend durch eine sichtbare, tatkräftige Beteiligung, d.h. Mitglieder arbeiten vorrangig mit den anderen zusammen. Endnutzer dagegen nutzen die Software und bringen nur sporadisch Arbeitspakete ein. Sie werden nicht als Mitglied anerkannt, da keine längere Bindung und Verpflichtung eingegangen wird.

- Der Teilprozess *Arbeitsplanung* besteht aus verschiedenen Unterprozessen wie „Zieldefinition“, „detaillierte Entwicklungsplanung“ und „Arbeitsverteilung“. Die Arbeitsmarktfunktion und das Wasserfall- oder evolutionäre Modell blicken dabei auf verschiedene Bereiche dieses Teilprozesses, die aber zusammengehören, da mit einer Planung eine spätere Arbeitsverteilung einhergeht und umgekehrt (s.o.).

Bei der *Zieldefinition* werden organisations- bzw. arbeitsprozessübergreifende Ziele festgelegt. Anders formuliert wird über mögliche Ideen/Ziele der Organisation bzw. des übergreifenden Arbeitsprozesses entschieden. Ausgehend von der Unterscheidung im Wasserfall- oder evolutionären Modell unterscheidet sich die Zieldefinition von der *detaillierten Entwicklungsplanung* im Detaillierungsgrad, d.h. bei der Zieldefinition werden die groben Ziele und Ideen, bei der Entwicklungsplanung die konkreten, zum jeweiligen Arbeitspaket bezogenen Ziele/Alternativen auf die eine oder andere Weise festgelegt. Genau genommen werden Alternativen mit groben Zielrichtungen bei der Zieldefinition generiert, aus denen eine ausgewählt und dann bei der detaillierten Entwicklungsplanung in kleine Zwischenziele/Schritte heruntergebrochen wird. Diese Planungstätigkeiten gibt es implizit bzw. in gewissem Ausmaß auch im Open Source-Entwicklungsmodell, was, wie oben angedeutet, aber nicht genauer untersucht wurde. Das parallele Arbeiten erschwert dabei die Zuordnung.

Die *Arbeitsverteilung* kann weiter in die Allokation bzw. Zusammenführung von Aufgabenbereichen/-paketen oder von Posten mit spezifischen Aufgaben-/Verantwortungsbereichen zu Mitarbeitern unterteilt werden. Die Allokation von temporären Arbeitsaufgaben bzw. eher unbefristeten Arbeitsbereichen zu Mitarbeitern bezieht sich auf die konkrete Softwareentwicklung. Spezifische Aufgaben-/Verantwortungsbereiche werden durch eine Postenbesetzung übernommen. Bei den Posten handelt es sich weitestgehend um andere, unbefristete Arbeiten bzw. Aufgaben außerhalb der Softwareentwicklung, die die Softwareentwicklung unterstützen oder beeinflussen können, wie Pflege der IuK-technischen Infrastruktur, Werbemaßnahmen, administrative Tätigkeiten/Vereinsvorsitz etc. Die Postenbesetzung ist zwar mit einer Arbeitsverteilung verbunden, doch wird auch hier kein direkter Einfluss auf die Softwareentwicklung genommen. Es handelt sich um eine Verteilung von z.T. nicht vollkommen klar spezifizierten Bündeln von Teilaufgaben. Zudem bestehen nur eine kleine Anzahl von Posten, die eher selten besetzt werden. Der Teilprozess „Arbeitsverteilung durch Postenbesetzung“ ist daher ein Bereich des Teilprozesses „Arbeitsverteilung von Arbeitspaketen“, der wegen seines andersartigen Kerns zusätzlich untersucht wird. Arbeitsverteilungen treten im Open Source-Entwicklungsmodell auch auf.

- Die *Leistungserstellung* beinhaltet die konkrete Transformation eines individuellen Arbeitsvermögens einer Arbeitskraft in ein konkretes Arbeitspaket, was in diesem Fall die Bearbeitung von Arbeitspaketen bzw. das Programmieren eines Quellcodeabschnitts ist. Sie bezieht sich auf die Implementierung eines detailliert geplanten Entwurfs im Wasserfallmodell. Beim evolutionären Modell wird ein erster geplanter Entwurf in dieser Funktion ständig überarbeitet (s.o.). Beim Open Source-Projekt hat der Entwurf eigentlich keine ho-

he Bedeutung, da der Entwurf meistens unstrukturiert z.B. im Kopf und in einem Versuchs- und Irrtumsprozess erfolgt (vgl. Jörgensen 2001; Mockus et al. 2002).

- Schließlich erfolgt die *Arbeitszusammenführung* der einzelnen Arbeitspakete der Mitarbeiter zu einem gemeinsamen Gesamtprodukt als abgeschlossene Arbeitsleistung. Die Arbeitszusammenführung hat große Ähnlichkeit mit der Abnahme- und Einführungs- bzw. Testphase im Wasserfallmodell, in der das Programm an einem bestimmten Termin auf seine geplanten Funktionen und seine Fehlerlosigkeit hin überprüft wird, womit die Arbeit der Softwareentwicklung enden kann. Beim evolutionären Entwicklungsmodell wird eine solche Abnahmephase, ebenso wie eine Wartungs-/Pflegephase implizit angenommen. Die Wartungs-/Pflegephase kann aber nicht zur Arbeitszusammenführung gezählt und muss ausgenommen werden, da es sich bei den Wartungsarbeiten nur um nachträgliche Reparaturen handelt und nicht um eine gemeinsame Anstrengung mit der Vereinigung von einzeln erstellten Arbeitspaketen.

Die Teilprozesse Leistungserstellung und Arbeitszusammenführung sind im Open Source-Entwicklungsmodell nicht eindeutig zuzuordnen. Im Prinzip gibt es eine Leistungserstellung bei der Erstellung von Entwicklungsrelease und Produktionsrelease, da in beiden Fällen Quellcode entwickelt wird. Genauso ist es bei der Arbeitszusammenführung: die Arbeitspakete werden zu einem dynamischen und unfertigen Entwicklungsrelease und zu einem stabilen Endprodukt als Produktionsrelease zusammengeführt. Leistungserstellung und Arbeitszusammenführung erfolgen also im Allgemeinen parallel im Open Source-Entwicklungsmodell, wobei aber Tätigkeiten, wie Programmieren, Testen, Releaseerstellung etc., eher dem einen oder dem anderen zugeordnet werden können (vgl. Jörgensen 2001: 328ff.; Mockus et al. 2002). Für eine Zuordnung der Teilprozesse Leistungserstellung und Arbeitszusammenführung kann an die zwei aufeinanderfolgenden Phasen der Erstellung von Entwicklungsrelease und Produktionsrelease angeknüpft werden. In der kreativen Phase des Entwicklungsreleases werden neue Softwarefunktionen bzw. umfangreiche Programmteile programmiert, weswegen die Leistungserstellung überwiegt. Während der Erstellung des Produktionsreleases steht die Arbeitszusammenführung der Arbeitspakete zu einer gemeinsamen Arbeitsleistung im Vordergrund. Zur Vereinfachung der Diskussion werden damit die Leistungserstellung mit der dynamischen, freien Phase und die Arbeitszusammenführung mit der Phase des stabilen, offiziellen Releases gleichgesetzt, da die abgeschlossene Arbeitsleistung im Open Source-Entwicklungsmodell den Endpunkt einer organisationalen Transformation von Arbeitskraft in Arbeitsleistung darstellt und einer Transformation in einem Unternehmen nahe kommt.<sup>20</sup>

## 2.4. Zusammenfassung

Neue Arbeitsformen wie die virtuellen Arbeitsnetze mit den Open Source-Projekten entstehen. Zugleich differenziert sich die Erwerbsarbeit aus und andere Arbeitsformen wie Eigenarbeit, Schwarzarbeit und Ehrenamtsarbeit erhalten eine zunehmende Bedeutung. Gerade neue Arbeitsformen entstehen im Zusammenhang mit der Verbreitung vernetzender und vernetzter IuK-Technik. Ein besonders interessanter Aspekt in der Diversifizierung gesellschaftlicher Arbeit(sformen) sind innerhalb der virtuellen Arbeitsnetze die Open Source-Projekte als neue Arbeitsform.

---

<sup>20</sup> Wartungs-/Pflegearbeiten werden indirekt ausgeführt und daher nicht betrachtet, da sie durch die ständige Beteiligung von Endnutzern bzw. von Softwareentwicklern als Nutzer sowie einem direkten Einbezug von Rückmeldungen in die laufende, kontinuierliche Softwareentwicklung integriert werden (vgl. Jörgensen 2001; Mockus et al. 2002).

In virtuellen Arbeitsnetzen arbeiten autonome Akteure temporär und informal zusammen und erstellen in diesem Fall ein gemeinsames Softwareprodukt, wobei zeitlich und örtlich ungebunden überwiegend über IuK-Technik kommuniziert und produziert wird. Bei der ablaufenden Softwareentwicklung werden nicht Phasenmodelle, Arbeitsstile und Projekttypen analysiert, sondern die Normen in bestimmten Situationen des Arbeitsprozesses untersucht.

Bisher steht die Erwerbsarbeit auch bei der Analyse von IuK-technisch vernetzter Arbeit im Zentrum der wissenschaftlichen Betrachtung. Am weitesten entfernt von der Erwerbsarbeit im Unternehmen ist die freiwillige virtuelle Arbeit in einem Open Source-Projekt, die eher mit Eigenarbeit und Ehrenamtsarbeit verbunden ist. Die Freiwilligkeit macht den Arbeitsprozess und das Arbeitshandeln besonders interessant, da sich Arbeitsprozess und Arbeitshandeln wahrscheinlich von der virtuellen Erwerbsarbeit bzw. Telearbeit im Unternehmen unterscheiden. Das Phänomen Open Source-Projekt mit seinem Arbeitsprozess ist daher in dieser Arbeit der zentrale Untersuchungsgegenstand. Es wird aber kein Vergleich zwischen dem Phänomen Open Source-Projekt mit der virtuellen Erwerbsarbeit bzw. Telearbeit im Unternehmen angestrebt.

Die IuK-Technik wurde betrachtet, da die virtuellen Arbeitsnetze kaum ohne eine umfassende und ständige Nutzung derselben zu verstehen sind. Die Fragestellung, wie die IuK-Technik die Arbeit in virtuellen Arbeitsnetzen beeinflusst, wurde anhand einer technikphilosophischen und –soziologischen Perspektive reformuliert und präzisiert sowie an die Softwareentwicklung angepasst. Die IuK-Technik stellt Software und Hardware des Internets und der Computer der Arbeitsnetzmitglieder dar. Sie ist ein Gestell bzw. eine gekapselte Black Box und damit ein Medium zur Welterschließung, mit dem sowohl kommuniziert und interagiert als auch produziert werden kann. Beim Arbeitsprozess der Softwareentwicklung bestehen daher gekapselte Artefakte, auf deren Basis ein Softwareprodukt erstellt wird. Das Internet, auf das vom Arbeitsnetz ausgehend nicht zugegriffen werden kann, ist dabei als gekapseltes Artefakt zu betrachten. Bei der Softwareentwicklung wird außerdem einerseits Quellcode programmiert, andererseits auf lauffähige Programme in kompiliertem Maschinencode als gekapselte Artefakte zur Unterstützung der Zusammenarbeit und Softwareentwicklung zurückgegriffen. Damit läuft der Arbeitsprozess weitestgehend über eine IuK-technische Infrastruktur ab, die in eine Kommunikations-, Informations- und Produktionsausrichtung eingeteilt werden kann. Die Infrastruktur kann also direkt an neue Normen angepasst werden. Folglich determiniert die verwendete IuK-technische Infrastruktur als gekapseltes Artefakt den Arbeitsprozess mit seinen Normen nicht, kann aber spezifische Normen unterstützen. Es ergibt sich die reformulierte Frage, wie die IuK-Technik als eher statische, gekapselte IuK-technische Infrastruktur den Arbeitsprozess der Softwareentwicklung mit seinem Arbeitshandeln beeinflusst. Da die Software als Maschinencode genutzt, die gleiche Software jedoch auch bearbeitet und weiterentwickelt werden kann, besteht eine Wissensspirale, bei der Wissen direkt durch neues Wissen angeichert werden kann. Da die Softwareentwicklung nicht determiniert ist, können verschiedene Interessensgruppen aus der Umwelt eines Arbeitsnetzes Einfluss auf diese nehmen. Die Endnutzer haben eine Beziehung über das Softwareendprodukt zur Softwareentwicklung, weswegen diese bei der Betrachtung der Softwareentwicklung mituntersucht werden. Die Beziehung zwischen Softwareentwicklern als Arbeitende und (End-)Nutzern wird als Nebenfrage untersucht. Weitere Interessensgruppen werden bei der Softwareentwicklung fürs erste vernachlässigt.

Der Arbeitsablauf bzw. -prozess bei der Softwareentwicklung steht im Zentrum dieser Arbeit, weswegen eine punktuelle bzw. situative Unterteilung in Form von Funktionen oder zeitlichen Phasen herangezogen wird, um Teilprozesse abgrenzen zu können. Bei ökonomischen Betrachtungen des Arbeitsmarkts wird vielfach ein implizit angenommener Ar-

beitsprozess durch mehrere Funktionen beschrieben, wie „Allokation“, „Motivation“, „Leistungserstellung“ und „Kontrolle“. Allokation bezieht sich auf den Eintritt in eine Organisation (Zusammenkommen von Arbeitssuchenden und Stellen/Aufgabenbereichen), aber auch die Arbeitsverteilung (Übernahme von konkreten Arbeitsaufgaben). Motivation und Leistungserstellung sind weitestgehend an die konkrete Transformation von (individueller) Arbeitskraft in Arbeitsleistung geknüpft. Die Funktion Kontrolle steht überwiegend mit der Zusammenführung aller (individuellen) Arbeitspakete zu einem Produkt in Zusammenhang. Demgegenüber wurde zur Verfeinerung an den Phasenmodellen der Softwareentwicklung aus der Informatik angeknüpft, wobei verschiedene Phasenmodelle wie Wasserfall- und evolutionäres Modell herangezogen wurden. Dabei wurde einerseits auf Gemeinsamkeiten<sup>21</sup> zwischen Phasenmodellen und Funktionen für eine Entwicklung von Teilprozessen der Softwareentwicklung geachtet, andererseits Unterschiede zur Detaillierung der Teilprozesse der Softwareentwicklung genutzt.

Funktionen und Phasen zusammengenommen beschreiben verschiedene Teilprozesse, die den idealisierten Arbeitsprozess der Softwareentwicklung bestimmen. Teilprozesse der Softwareentwicklung sind „Eintritt/Rekrutierung“, „Arbeitsplanung mit Zieldefinition, detaillierter Entwicklungsplanung und Arbeitsverteilung“, „Leistungserstellung“ sowie „Arbeitszusammenführung“. Die Arbeitsverteilung bezieht sich zum einen auf eine Arbeitsverteilung von Arbeitspaketen/-bereichen bei der Softwareentwicklung und zum anderen auf eine Arbeitsverteilung von Posten, die Aufgaben außerhalb der Softwareentwicklung umfassen, aber die Softwareentwicklung unterstützen. Diese Teilprozesse stellen den Analyserahmen dar, in dem untersucht wird, wie die Normen und Regeln das Arbeitshandeln bestimmen. Da das Open Source-Projekt Untersuchungsgegenstand dieser Arbeit ist und das Phasenmodell des Open Source-Entwicklungsmodells auf einer parallelen Bearbeitung basiert, wurden bei der Definition des idealisierten Arbeitsprozesses die einzelnen Bearbeitungsschritte des Open Source-Entwicklungsmodells mit diesem neuformulierten Arbeitsprozess abgeglichen und verfeinert.

Ausgehend von der zusätzlichen Betrachtung der IuK-Technik interessiert der Zusammenhang zwischen IuK-technischer Infrastruktur und Arbeitsprozess, weswegen ein Bezug zwischen der IuK-technischen Infrastruktur und den Teilprozessen hergestellt wird.

---

<sup>21</sup> Der Teilprozess „Planung der Softwareentwicklung“ hängt beispielsweise mit der Arbeitsverteilung zusammen. Der Teilprozess „Implementierung/Entwicklung der Software“ stimmt weitestgehend mit der Leistungserstellung überein.

### 3. Koordination und Arbeitsnetze als Organisationsform – Stand der Forschung und (Neu-)Definition

Der Arbeitsprozess in einem Open Source-Projekt wird untersucht. Dabei interessiert, welche Normen bzw. Institutionen<sup>22</sup> diesen Arbeitsprozess strukturieren. Das Open Source-Projekt gehört als Arbeitsnetz (s. Kap. 2.1.) zur allgemeineren Organisationsform Netzwerk, woraus sich die Notwendigkeit ergibt auf den Netzwerkdiskurs in der sozial- und wirtschaftswissenschaftlichen Literatur einzugehen. Dabei gibt es nicht den einen Netzwerkdiskurs, sondern drei Strömungen mit unterschiedlichem Blickwinkel auf Netzwerke (vgl. Sydow 1992: 118ff.; Weyer 2000: 14ff.; Powell, Smith-Doerr 1994; Windeler 2003). Erstens ist der Netzwerkdiskurs über Relationen zwischen spezifischen Akteuren (methodische Netzwerkanalyse; s. Kap. 3.1.1.) und zweitens über empirische Erscheinungsformen (s. Kap. 3.1.2.) zu nennen. Drittens besteht ein Netzwerkdiskurs in einer umfangreichen Theoriediskussion (s. Kap. 3.1.3.), die Netzwerke neben den Organisationsformen Markt und Unternehmen oder Staat als „Hierarchie“ verortet. Als Organisationsformen werden in einer vorgreifenden Definition künstlich geschaffene Institutionen wie Markt, bürokratische Organisationen (wie Unternehmen, Staat) und Netzwerk gesehen, die ein spezifisches Ziel verfolgen.

Die methodische Netzwerkanalyse ist hier nicht von großer Bedeutung, wird aber vollständigshalber vorgestellt, da einige ihrer Überlegungen implizit in der empirischen und theoretischen Literatur zu finden sind. Im empirischen und theoretischen Diskurs wurde die Organisationsform (1) Markt als Metapher für Wettbewerb, (2) Unternehmen bzw. Staat als Metapher für Hierarchie und (3) Netzwerke als Metapher für Kooperation<sup>23</sup> gesehen und überwiegend miteinander gleichgesetzt. Damit geht der theoretische Diskurs über Organisationsformen mit einer Steuerungs- bzw. Koordinationsdebatte einher (vgl. die Darstellung des Diskursverlaufs aus einer politologischen Perspektive von Wiesenthal 2005).

Unter Koordination wird als vorgreifende Definition eine Regulation oder Steuerung verstanden, die durch sogenannte Koordinationsformen, d.h. durch einen fokalen Akteur als hierarchisches Element, durch gleichgestellte Akteure mit kooperativer Aushandlung oder durch einen Wettbewerb der Akteure erfolgt. Da die Netzwerkdiskussion gerade in ihrer Auseinandersetzung mit Markt und bürokratischer Organisation (Unternehmen/Staat) für die Koordinationsdiskussion zentral ist, wird die Koordinationsdiskussion mit Hilfe der Netzwerkdiskussion aufgerollt.

Ein Ergebnis des Literaturüberblicks über die empirische und theoretische Debatte ist (s. Kap. 3.1.4.), dass die Organisationsform Netzwerk, aber auch die anderen Organisationsformen Markt und Unternehmen/Staat, von ihrer Koordination durch Koordinationsformen zu trennen sind. Koordinationsformen wie Kooperation, Hierarchie/Macht und Wettbewerb sind dabei Normen bzw. Institutionen auf der Handlungsebene unter der Organisationsformenebene. In allen Organisationsformen treten wettbewerbliche, hierarchische und kooperative Elemente auf.

Damit wird eine für die Ökonomie und Soziologie unübliche Perspektive gewählt. Eine gebräuchliche Sichtweise ist beispielsweise, sowohl Hierarchie als auch Wettbewerb als Ko-

---

<sup>22</sup> Normen und Institutionen werden normalerweise begrifflich in einem Punkt unterschieden. Während der Begriff Normen sehr allgemein alle Verhaltensregeln beinhaltet, werden Institutionen üblicherweise als dauerhafter und statischer betrachtet. In dieser Arbeit werden Institutionen als Bündel von Normen bzw. Regeln verstanden.

<sup>23</sup> s. z.B.: <http://www.ifg-muenster.de/toplinks/glossar/glossar.php?begriff=29>

ordinationsmechanismen zu sehen, um eine Kooperation herbeizuführen (vgl. Coase 1937; Williamson 1985). Auch wird in der Spieltheorie eher zwischen Kooperation und Wettbewerb unterschieden, wobei Hierarchie/Macht nicht betrachtet wird (vgl. Holler, Illing 1996).

Diese Trennung zwischen Organisation und Koordination oder zwischen Organisationsform und Koordinationsform hat folgende Vorteile. Zum einen kann die Unterfrage nach dem Einfluss der IuK-Technik auf den Arbeitsprozess der Softwareentwicklung mittels der Koordinationsformen in den untersuchten Arbeitsnetzen wie bei Becher et al. (2007) untersucht werden. Zum anderen können die Koordinationsformen als Normen bzw. Institutionen beim Arbeitsprozess analysiert werden, da in dieser Arbeit die strukturierenden Normen bzw. Institutionen beim Arbeitsprozess der Softwareentwicklung eines Netzwerks in Form eines Arbeitsnetzes im Fokus stehen. Folglich kann in der empirischen Untersuchung eine Kombination aus Kooperation, Hierarchie/Macht und Wettbewerb bei der Softwareentwicklung in Arbeitsnetzen festgestellt werden, wobei auch eine überragende Dominanz einer Koordinationsform auftreten könnte.

Die theoretische Literatur zu Netzwerken und Koordination aber leidet unter einem Begriffswirrwarr, da verschiedene Disziplinen ihre eigenen Begrifflichkeiten setzen (vgl. Wolf 2000: 96; Windeler 2003: 54; Rölle, Blättel-Mink 1998: 66; Wiesenthal 2005: 232; Krebs, Rock 1994). Um diesem Begriffsdurcheinander zu entgehen, wird in Kapitel 3.2. der Begriff der „Organisation“, „Organisationsformen“, „Koordination“ und „Koordinationsformen“ ausgehend vom Institutionenbegriff definiert. Die Definition erfolgt auch, da in diesem Fall ein neuer Weg mit der Trennung zwischen Organisations- und Koordinationsform gegangen wird. Damit der Arbeitsprozess und seine Koordination untersucht werden können, wird dabei auch ein Analyseinstrumentarium gebildet. Dabei werden die Koordinationsformen im Hinblick auf den zu untersuchenden Arbeitsprozess definiert. Die Definition der Koordinationsformen ist auch ein Kompromiss zwischen Anpassung an das Untersuchungsobjekt Netzwerk einerseits und andererseits der Integration aller Konzepte der Koordinationsformen aus verschiedenen Disziplinen.

Im Folgenden wird als erstes die Literatur zu den Netzwerken bzw. zur Koordination betrachtet und die Trennung zwischen Organisationsform und Koordinationsform begründet. Danach erfolgt die Definition zum einen von Organisation und Organisationsformen, wobei das Open Source-Projekt unter der allgemeinen Organisationsform Netzwerk gefasst wird. Zum anderen werden Koordination und besonders Koordinationsformen systematisch nach einem Kategorisierungsschema auf einer Ebene begrifflich bestimmt.

### **3.1. Diskurs über Koordination und Organisation aus der Netzwerkperspektive – Abgrenzung von Koordinationsform und Organisationsform**

Netzwerke sind in den letzten zwei Jahrzehnten immer stärker ins Zentrum des wissenschaftlichen Diskurses gerückt. Dabei hat der Terminus Netzwerk von den 1960er bis in die 1990er Jahre einen Bedeutungswechsel erfahren. Wurde in den 1960er Jahren noch der Netzwerkbegriff als illegitim und intransparent im Sinne von Vetternwirtschaft/Seilschaften angesehen, also negativ konnotiert<sup>24</sup>, wurden dagegen in den 1990er Jahren Netzwerke als legitim und wünschenswert betrachtet. Die Verbreitung der neuen, vernetzten Informations- und Kommunikationstechnologien unterstützte die Verbreitung

---

<sup>24</sup> Negativ konnotierte Netzwerke wie Vetternwirtschaft/Seilschaften werden in dieser Arbeit nicht untersucht.

des Netzwerkbegriffs in verschiedenen Disziplinen. Mittlerweile ist der Netzwerkgedanke nicht nur in den Wissenschaften verbreitet, sondern hat auch Eingang in die wirtschaftliche Sphäre in Form von Projekten und als eine „projektbasierte Rechtfertigungsordnung“ gefunden (Boltanski, Chiapello 2003: 191ff.).

In sozial- und wirtschaftswissenschaftlichen Disziplinen steht größtenteils die Entstehung von Unternehmensnetzwerken im Fokus (vgl. Wiesenthal 2005: 223ff.). Die krisenhaften, sich häufig ändernden sozio-ökonomischen Entwicklungstendenzen, ausgelöst durch eine neue Unmittelbarkeit des Marktprinzips in einer globalisierten Welt, das Auftreten des Internets und die Krise der bürokratischen hierarchischen Steuerung im Unternehmen mit ausgedünnten Unternehmenshierarchien und Outsourcing führten zu einer verstärkten Vernetzung zwischen Unternehmen (vgl. Sauer 2001: 30ff.; Hirsch-Kreinsen 2002: 106ff.; Hillmann 1994: 332; Schmiede 2006: 466f.). Zusätzlich werden die Konflikte zwischen gesellschaftlichen Teilsystemen und stringent verfolgten Handlungslogiken der Organisationen bzw. Unternehmen für das verstärkte Auftreten von Unternehmensnetzwerken verantwortlich gemacht (Schimank 2005: 39ff.). Die Netzwerkmetapher wird in diesem Zusammenhang auch verwendet, um widersprüchliche Anforderungen in ein „Handlungsmodell“ (Netzwerk) zu integrieren, so „dass sie, wenn nicht auflösbar, so zumindest konstruktiv bearbeitbar werden“ (Steger 2003: 10).

Diese breite Diskussion führt zu einem eklektischen Diskurs, bei dem weitestgehend nicht an Begriffe und Überlegungen zwischen Disziplinen und in der eigenen Disziplin angeknüpft wurde (Wiesenthal 2005: 226). „Der Netzwerkdiskurs präsentiert sich als dickes Knäuel heterogener Bedeutungen, einer unüberschaubaren Vielfalt von Netzwerkbegriffen, -theorien und –analyseansätzen“ schreibt Wolf (2000: 96; vgl. auch Rölle, Blättel-Mink 1998: 66; Windeler 2003: 54). Diese irritierende Fülle an Netzwerkbegriffen hängt mit den verschiedenen Entstehungskontexten und Anwendungsbereichen in Ökonomie, Soziologie, Sozialanthropologie und Politikwissenschaften zusammen (s.u.; vgl. z.B. Wiesenthal 2005; Steger 2003).

Für eine Strukturierung des Netzwerkdiskurses (inkl. seines Koordinationszusammenhangs) werden, wie oben angekündigt, hier drei verschiedene, analytisch trennende Sichtweisen von Netzwerken herangezogen: eine methodische, eine theoretische und eine empirische Sicht. Die analytische Trennung konzentriert sich auf den Hauptfokus der drei Sichtweisen, da sich z.B. die Netzwerktheorie oft des empirischen Diskurses bedient und umgekehrt. Auch bei der methodischen Netzwerkanalyse spielt ein theoretisches Vorverständnis eine Rolle oder es werden theoretische Annahmen herangezogen. Die Diskurse werden nicht vollständig, aber hinsichtlich der zentralen Themen vorgestellt, da diese Diskurse sehr ausufernd sind und bereits einige Autoren diese sehr umfangreich dargestellt haben (s. z.B. Wiesenthal 2005; Müller-Jentsch 2003; Hirsch-Kreinsen 2002; Weyer 2000; Krebs, Rock 1994).

Beginnend mit der methodischen, formalen Netzwerkanalyse werden dann die Netzwerke in ihren empirischen Erscheinungsformen vorgestellt. Daraufhin folgt ein Überblick über die umfangreichere Theoriediskussion. Am Schluss wird auf neuere Entwicklungen hingewiesen und für eine getrennte Betrachtung zwischen Organisationsformen und das Auftreten von Kooperation, Hierarchie und Wettbewerb plädiert.

#### **3.1.1. Methodische, formale Netzwerkanalyse**

Die umfassendste Definition von Netzwerken bezeichnet Netzwerke als „a set of nodes and the pattern of ties among such nodes“ (Zuckerman 2003: 546). Die Definition ist so allgemein, dass alle Beziehungen zwischen Menschen in Märkten genauso wie in Unternehmen oder anderen Beziehungskonstellationen darunter fallen. Damit ist sie eine grund-

legende Überlegung zu den Organisationsformen Markt, Unternehmen und Netzwerk. Diese Definition beinhaltet (wirtschaftliches wie politisches) Handeln, das damit sozial eingebettet und auf einer Ebene erscheint (Sydow, Windeler 2000: 3). Diese allgemeine Definition ist also der kleinste gemeinsame Nenner in der Diskussion über Organisationsformen, wie Netzwerke.

Die methodische, formale Netzwerkanalyse untersucht von dieser Definition ausgehend die Beziehungen zwischen Akteuren. Dabei werden i.A. spezifische Beziehungsstrukturen des menschlichen Beziehungsgeflechts vor allem zwischen Individuen erhoben. Der zentrale Gedanke der Netzwerkanalyse ist, dass soziale Akteure in einem Geflecht sozialer Beziehungen stehen, die ihre eigenen Handlungsmöglichkeiten sowohl erweitern als auch einschränken, und die ihre Wahrnehmungen, Einstellungen und Handlungen beeinflussen (vgl. Trezzini 1998: 537). Fokus der Untersuchungen sind eher bestimmte natürlich vorkommende soziale Systeme, wie Familien, Cliquen oder Kommunen (vgl. Jansen 1999). Beziehungen zwischen Organisationsformen wie Unternehmen können Fokus der Netzwerkanalyse sein, wobei z.B. die Mitgliedschaft einer Person in den Aufsichtsräten von zwei oder mehr Unternehmen als Merkmal verwendet wird. Aber auch andere Beziehungsstrukturen können damit untersucht werden, wobei irrelevant ist, ob eine Beziehung in einem Unternehmen, im Markt oder in anderen Beziehungsgeflechten stattfindet (vgl. Steger 2003).

Die Beziehungen beinhalten verschiedene inhaltliche Beziehungsdimensionen, d.h. sie können z.B. formaler (vertraglicher) oder informaler (nicht-vertraglicher) bzw. einfacher (eindimensionaler) und multiplexer (mehrdimensionaler) Natur sein. Für den Erkenntnisgewinn werden anfangs eine oder mehrere Beziehungsdimensionen ausgesucht. Im Vordergrund der Netzwerkanalyse steht dann die sozialwissenschaftliche Methode der Datenbeschaffung und Datenauswertung von diesen Beziehungen.<sup>25</sup> Damit sollen spezifische Strukturen mittels quantitativer statistischer Verfahren festgestellt werden (vgl. Jansen 1999). Die Ausrichtung auf die Beziehungsstrukturen und quantitative statistische Verfahren lässt nur indirekt und u.U. unter Hinzuziehung weiterer Informationen über die untersuchten Akteure Schlüsse auf Normen oder Werte zu. Die Normen bestimmen also die Beziehungsstrukturen und werden durch die Betrachtung der inhaltlichen Dimension von Interaktionen sichtbar gemacht (vgl. Zuckerman 2003).

Die formale Netzwerkanalyse wird als Instrumentarium bzw. als Methode zur Untersuchung der Beziehungsnetzwerke verwendet. Es besteht ein gewisses Theoriedefizit (vgl. Trezzini 1998: 515), doch ist die formale Netzwerkanalyse nicht vollkommen theorieles: Die Theorie bezieht sich zum einen auf eine handlungstheoretische Grundlegung der Netzwerkanalyse. Zum anderen werden zu untersuchende Indizes mit dem Begriff des Sozialkapitals, d.h. dem Grad an sozialen Beziehungen eines Individuums, verknüpft (vgl. Jansen 1999: 16ff.; 159ff.). Außerdem gehen auch machttheoretische Überlegungen in dieses Konzept ein, die eine der wenigen Thematisierungen einer Koordinationsform (s.u.) darstellen.

Organisationsformen oder Koordinationsformen wurden bei der methodischen Netzwerkanalyse, wie im Folgenden gezeigt, kaum dezidiert untersucht. Ein diskursiver Austausch mit dem folgenden empirischen und theoretischen Diskurs besteht nicht, manchmal scheint bei diesen aber ein netzwerkanalytisches Grundverständnis zu bestehen. Aber die Koordinationsformen Macht, Kooperation und Wettbewerb treten in der Netzwerkanalyse in anderen Zusammenhängen auf, werden aber nicht im Zusammenhang mit Koordination diskutiert.

---

<sup>25</sup> In der Anwendung bestehen Ähnlichkeiten zur mathematischen Graphentheorie (vgl. Jansen 1999).

Ein solcher Zusammenhang besteht nach Burt (1992) bei strukturellen Löchern, also Nicht-Beziehungen zwischen Clustern, die als Grenzen zwischen gehäuften Relationen interpretiert werden können. Spezifische Nicht-Beziehungen lassen auf Normen bzw. Institutionen schließen: Anhand von Beziehungsstrukturen mit strukturellen Löchern werden Wettbewerbs- (Märkte), Kooperations- (Cliques) und hierarchische Beziehungen festgestellt (Burt 1992: 39), also durch eine negative und nicht durch eine positive Definition. Ähnliche sozialpsychologische Forschung drehte sich um die Gruppenproduktivität, die bei sternförmigen (hierarchisch), netz-/kreisförmigen (kooperativ) oder anderen Kommunikationsstrukturen auftrat (vgl. Wilke, van Knippenberg 1992).

Eine weitere Ähnlichkeit zu Organisations-/Koordinationsformen haben „four basic flows in the organizational context: (1) exchange of goods; (2) affect and liking (expressive); (3) information and ideas (cognitive), and (4) influence and power (prescriptive)“, die verschiedene inhaltliche Beziehungsdimensionen darstellen (Tichy, Fombrun 1979: 925). Diese inhaltlichen Beziehungsdimensionen können u.U. als Organisations- oder Koordinationsformen interpretiert werden.

Dieser Netzwerkauffassung, die vor allem auf der Mikroebene verankert ist, steht eine Netzwerkauffassung gegenüber, die ein Netzwerk als Organisationsform auf der organisationalen Ebene (Mesoebene) als eigene Entität betrachtet. Auf Sie wird im Folgenden eingegangen.

#### **3.1.2. Netzwerke im empirischen Diskurs**

##### *3.1.2.1. Ökonomische, soziologische und politikwissenschaftliche Diskurse*

Nicht nur Managementtheoretiker (Kühl 1999), sondern auch Soziologen und Wirtschaftswissenschaftler sahen verstärkt Interorganisations- bzw. Unternehmensnetzwerke entstehen. Die Netzwerkdiskussion hat sich auch im politikologischen Diskurs über politische Inhalte (Policy) niedergeschlagen, wobei zunehmende Politiknetzwerke mit informalen Beziehungen entdeckt wurden (vgl. z.B. Weyer 2000; Steger 2003; Williamson 1985, 1991; Schimank 2005: 38ff.).

Die verbreitete Diskussion über Netzwerke erfolgte vor allem in der empirischen Netzwerkforschung und einem theoretischen Netzwerkdiskurs. Beide Diskussionen sind miteinander verknüpft. Eine Gemeinsamkeit beider Diskurse ist eine gleiche Netzwerkdefinition: Netzwerke wurden zielgerichtet gegründet, sind also Organisationsformen. Sie sind keine natürlichen, d.h. alltäglichen Beziehungen, wie Nachbarschaftskontakte, Freundes- und Bekanntenkreis o.ä., aus denen diese künstlichen Netzwerke aber entstehen können (Steger 2003). Ausgehend von diesem gemeinsamen Standpunkt werden in der Soziologie, Ökonomie und Politikologie empirische und theoretische Ergebnisse zu Netzwerken beigetragen.

##### *3.1.2.2. Verschiedene empirische Netzwerkformen*

Im empirischen Netzwerkdiskurs werden die Netzwerke als Organisationsformen betrachtet und es wird dabei untersucht, wie diese in verschiedenen Ausformungen und Merkmalen auftreten und z.T. wie diese funktionieren. Unter den untersuchten Netzwerken sind regionale (Akteurs-)Netzwerke (z.B. Heidenreich 2000; Schubert et al. 2001), strategische Allianzen/Netzwerke (z.B. Sydow 1992; Backhaus, Plinke 1990), Policy-Netzwerke (vgl. Pappi, König 1995; Marin, Mayntz 1991), Innovations-/Forschungs- und Entwicklungszentralen (vgl. Rammert 1997; Kowol 1998; Pfirrmann, Hornschild 1999) sowie Zulieferer-/Produktionsnetzwerke (z.B. Pohlmann 1996; Heidling 2000), um einige zu nennen (s.a. Weyer 2000; Müller-Jentsch 2003: 125ff.). Da auch bei der empirischen Netzwerkforschung

schung eine Fülle an Literatur publiziert wurde und sich diese Netzwerke z.T. überschneiden (z.B. regionale Innovationsnetzwerke), aber auch in Merkmalen unterscheiden (s.u.), wird ein Überblick anhand von bestimmten Merkmalen der untersuchten Netzwerke verschafft. Diese Elemente stellten sich bei Sichtung der Literatur heraus, sind aber auch an die Einteilung von Weyer (2000), und die Netzwerkanalyse mit Akteuren und Beziehungsdimension angelehnt. Dabei wurden spezifische Merkmale vielfach zur Charakterisierung von untersuchten Netzwerken herangezogen. So hat die empirische Netzwerkforschung Netzwerke mit verschiedenen Akteuren und Zielen in verschiedenen Gesellschaftsbereichen festgestellt. Im Zentrum stehen Interorganisationsnetzwerke, wobei oft von den konkreten Individuen abstrahiert wird und Organisationen bzw. Unternehmen die untersuchten Akteure mit spezifischen Beziehungen sind. Gerade Beziehungsgeflechte zwischen einzelnen Unternehmen werden häufig untersucht (vgl. Hirsch-Kreinsen 2002; Kowol, Krohn 1997).

Da die empirische z.T. mit der theoretischen Diskussion verknüpft ist, wird an dieser Stelle auf das Spezifische der empirischen Erkenntnisse eingegangen. Die empirische Netzwerkforschung bestätigt dabei die Merkmale, die die eher theoretische Netzwerkliteratur diskutiert (vgl. Kowol, Krohn 1997). Der empirische Netzwerkdiskurs wird also anhand der festgestellten Merkmale wie Akteure und Beziehungsdimensionen angelehnt an die methodische Netzwerkanalyse vorgestellt, wobei die Beziehungsdimensionen Ziele, informelle/formale Beziehungen und die räumlich-zeitliche Dimension umfasst. Die Merkmale sind nicht überschneidungsfrei, da z.B. regionale Netzwerke zusammen Forschung- und Entwicklung vorantreiben können.

#### 3.1.2.3. *Empirische Merkmale verschiedener Netzwerkformen*

Netzwerke bestehen aus gleichen und unterschiedlichen autonomen Akteuren, wobei die Akteure aus unterschiedlichen gesellschaftlichen Bereichen kommen können. Bestehen Zulieferer-/Produktionsnetzwerke fast ausschließlich aus Unternehmen, setzen sich Innovations- bzw. Forschungs- und Entwicklungsnetzwerke zwischen Unternehmen, staatlichen Organisationen, nicht-staatlichen Verbänden (NGOs) und Universitäten oder auch nur zwischen Unternehmen zusammen (vgl. Mahnkopf 1994: 75; Rammert 1997; Pfirrmann, Hornschild 1999; Pappi, König 1995; Marin, Mayntz 1991). Bei Fokussierung auf Unternehmensnetzwerke (z.B. Zulieferer-/Produktionsnetzwerke) können die Akteure nach ihrer Stellung in der Wertschöpfungskette unterschieden werden, d.h. sie kommen aus der gleichen oder unterschiedlichen Branchen bzw. Wertschöpfungskette oder sind der Wertschöpfungskette vor-, nachgelagert oder auf der gleichen Ebene angesiedelt (z.B. Pohlmann 1996; Pfirrmann, Hornschild 1999).

Betrachtet man die Organisationsgröße anhand der Beschäftigtenzahl vor allem in Unternehmen sind Netzwerke vorhanden, in denen nur Klein- und Kleinstunternehmen oder nur Großunternehmen vorkommen. Netzwerke von Klein- und Kleinstunternehmen bzw. Ein-Personen-Selbständigen sind z.B. Projekt-/Arbeitsnetzwerke im Medienbereich, Partnerschaften von Anwälten in Anwaltskanzleien oder (landwirtschaftliche) Kooperativen (vgl. Windeler et al. 2000; Wenny 1999; Reichwald et al. 1999; Blair 2003; Waters 1989; Lazeaga 2001). Auch Mischformen, d.h. Beziehungen zwischen Groß- und Klein- bzw. Kleinstunternehmen, treten auf. Dazu gehören auch (Unterstützungs-)Netzwerke oder Akteursnetzwerke mit „Privatpersonen“ und Unternehmen oder politischen Akteuren voluntaristischer Organisationen (Bluhm et al. 2003: 94f.; Mahnkopf 1994: 76f.; Krömmelbein, Schmid 2000: 230f.; Brand 2004; Schubert et al. 2001; Becher et al. 2005; Menard 2004: 348ff.; Pappi, König 1995; Marin, Mayntz 1991).

Eine Beziehungsdimension stellt die Realisierung gemeinsamer Vorteile bzw. die Verfolgung gemeinsamer Ziele dar, wobei eine Zusammenarbeit entsteht. Die Akteure von Innovations- bzw. Forschungs- und Entwicklungsnetzwerken verfolgen Innovations- oder Lernziele, wobei Informations- bzw. Wissensexternalitäten der Einzelnen mit Spezialwissen abgeschöpft werden (vgl. Rammert 1997; Kowol 1998). Netzwerkteilnehmer können aber auch nur über bestimmte Themen Erfahrungen austauschen oder Informationen über Angebot und Nachfrage weitergeben. Bei einer gemeinsamen Wertschöpfung bzw. Produktion z.B. bei Zulieferer-/Produktionsnetzwerken steht der Aufbau einer arbeitsteiligen Struktur und der unternehmerische Rückzug auf die Kernkompetenzen im Vordergrund. Strategische Allianzen/Netzwerke und Kartelle zielen auf eine Erhöhung der Erlöse z.B. durch Zugang zu neuen Märkten oder aufgrund größerer Marktmacht bei gleichzeitiger Risikominimierung. Die Steuerung von gesetzgeberischen Entscheidungs- oder Implementationsprozessen erfolgt in Policy-Netzwerken.

Die Netzwerke können auch vertraglich bzw. formal geregelt sein oder rein informal ablaufen (vgl. Hirsch-Kreinsen 2002; Menard 2004: 348ff.). Beispiele für eher vertragliche Netzwerke sind Joint Ventures bzw. Franchising, für informale Netzwerke Kartelle. Dabei können informale Netzwerke deckungsgleich mit z.T. persönlichen Unterstützungsnetzwerken sein, vertragliche Abmachungen als konkrete Kooperationen betrachtet werden (vgl. Bluhm et al. 2003).

Auch eine räumliche und zeitliche Dimension in Form von Regionalität und Internationalität spielt in den Netzwerkbeziehungen eine Rolle. Sind Policy-Netzwerke oftmals eher regional ausgerichtet (vgl. Pappi, König 1995; Marin, Mayntz 1991), so schließen sich multinationale Konzerne zu internationalen Allianzen zusammen (vgl. Lo 2001; Plehwe 2000). Die Akteure einer Region können sich aber auch zu Netzwerken zusammenschließen, um sich an wirtschaftlich schwierigen, aber auch hochdynamischen Standorten gegenseitig zu unterstützen (vgl. Heidenreich 2000; Schubert et al. 2001). Zeitlich unterscheiden sich Netzwerke über die Dauer der Zusammenarbeit. Projektnetzwerke sind durch temporäre, eher kurzfristige, strategische Allianzen/Netzwerke durch langfristige, z.T. vertraglich abgesicherte Beziehungen ausgezeichnet (Sydow, Windeler 2004).

Manche Netzwerke basieren vor allem auf Face-to-Face-Beziehungen, manche sind vor allem virtuell. Als ein Beispiel für virtuelle Netzwerke können virtuelle Unternehmen gelten (vgl. Sydow 1992: 118ff.; Kortzfleisch 1999; Mertens, Faisst 1996; Trautwein-Kalms 1997).

Weitere Netzwerke wie das zu untersuchende Arbeitsnetz Open Source-Projekt können als Innovationsnetzwerk zur Generierung innovationsrelevanten Wissens (vgl. Taubert 2006) als auch als virtuelles Netzwerk mit überwiegend IuK-technischer Kommunikation verstanden werden (s. Kap. 5).

#### **3.1.3. Idealtypisierte Netzwerke im theoretischen Diskurs - Stand der Forschung**

##### *3.1.3.1. Ökonomische, soziologische und politikwissenschaftliche Theorieströmungen*

Die empirischen Untersuchungen sind, wie oben dargestellt, mit dem disziplinären theoretischen Diskurs über idealtypisierte Netzwerke verbunden. Im theoretischen Netzwerkdiskurs wird aber versucht, anhand gemeinsamer inhaltlicher Beziehungsmerkmale Netzwerke als *idealtypische* Entitäten zu definieren und ihr Entstehen zu erklären. Dabei wird in dem eklektischen Diskurs das Netzwerk als Organisationsform mit eigenständigen Merkmalen bestimmt, wobei die Netzwerke auch in einer idealtypischen Betrachtung unterschiedlich definiert werden (s.u.; vgl. Wiesenthal 2005; Krebs, Rock 1994; Rölle, Blättel-Mink 1998: 66). Der folgende kurze Abriss des theoretischen Diskurses zeigt nur einen groben Ausschnitt, weitere Darstellungen der Definitionen und des Diskurses können z.B.

bei Wiesenthal (2005), Müller-Jentsch (2003) und Windeler (2001: 237ff.) nachgelesen werden.

Wie beim empirischen Diskurs bestehen beim theoretischen Diskurs ökonomische, politikwissenschaftliche und soziologische Strömungen, die mehr oder weniger aufeinander aufbauen (vgl. Wiesenthal 2005: 223ff.).

Die ökonomische Strömung basiert auf der Neuen Institutionenökonomie, deren zentraler Autor Williamson (1985, 1991) als erster Netzwerke als Hybrid zwischen Markt und Hierarchie feststellte. Unter Hierarchie wurde in der Neuen Institutionenökonomie Unternehmen verstanden (Weyer 2000: 8). Das Netzwerk als Hybrid hat dabei mittlere Ausprägungen von marktlichen und hierarchischen Elementen (Williamson 1991: 281; Sydow 1992: 98ff.; Picot et al. 1996; Krebs, Rock 1994: 325ff.; für eine Darstellung der neo-institutionalistischen Argumentation s. Göbel 2002).

Diese Diskussion hat sich auch im politologischen Diskurs niedergeschlagen, bei dem unter Hierarchie der Staat und unter Gemeinschaft das Netzwerk verstanden wird (vgl. Wiesenthal 2005; Streeck, Schmitter 1985). Auch die Autoren der soziologischen netzwerktheoretischen Strömung setzen an der ökonomischen Strömung an, kritisieren aber deren Sichtweise hinsichtlich der Netzwerkdefinition. Die Vertreter der soziologischen Strömung wie Powell (1996) sehen Netzwerke als eine eigenständige Organisationsform in Abgrenzung zu Markt und Hierarchie (s.a. Adler 2001: 219f.; Teubner 1996; Windeler 2001; Krebs, Rock 1994: 330ff.). Die Eigenständigkeit des Netzwerks wird gegenüber Markt und Unternehmen durch eigenständige Merkmale belegt. Unternehmensnetzwerke wurden auch aus systemtheoretischer Sicht (vgl. Teubner 1996) und strukturationstheoretischer Sicht (vgl. Windeler 2001) definiert. Ihre Eigenständigkeit wurde in beiden Theorien durch eine Rekombination von Markt- und Hierarchieelementen begründet, wobei jedes Netzwerkmitglied für sich und für das Netzwerk agiert.

Außerdem basiert die Definition des Netzwerks als eigenständige Organisationsform auf empirischen Beobachtungen (z.B. Mayntz 1992; Mahnkopf 1994).

Im theoretischen Netzwerkdiskurs ist es mittlerweile akzeptiert, Netzwerke als eigenständig in Abgrenzung zu Markt und Unternehmen/Staat als Hierarchie zu definieren. Im Weiteren wird auf die Netzwerkdefinition mit eigenständigen Merkmalen eingegangen.

#### 3.1.3.2. *Abgrenzung der Netzwerke von Hierarchien und Märkten*

Im theoretischen Diskurs sind Unternehmen/Staat bzw. formale Organisationen und Markt als Referenzpunkte unverrückbar und Ausgangspunkt der Diskussion. Es wurden Idealtypen dieser Organisationsformen entwickelt: Markt ist ein Spotmarkt mit spontanem und kurzfristigem Austausch autonomer Akteure, der über Preise koordiniert wird. Als ein weiteres Merkmal wird der Tausch genannt (vgl. Weyer, 2000; Windeler, Sydow 2004: 10f.).

Unternehmen bzw. Staat als Hierarchien haben für eine Zusammenarbeit langfristige Beziehungen mit den Organisationsmitgliedern und werden über formale Regelungen bzw. Routinen sowie Anweisungen gesteuert. Weise (2000: 181ff.) benennt dabei formale Regelungen bzw. Routinen als Organisationsmechanismus Norm, dagegen Dolata (2003) sieht Macht in einem „Technisierungsprozess“ wirken.

Diese eigenständigen Merkmale überschneiden sich nicht mit den eigenständigen Merkmalen der Netzwerke, wobei Weyer (2000) einige Merkmale der Organisationsformen zusammengetragen hat und miteinander vergleicht (s. Tab. 3-1). Netzwerke haben mittelfris-

tige informale Beziehungen zwischen Organisationen und werden z.B. über Vertrauen und Verhandlung gesteuert.<sup>26</sup>

Tab. 3-1: Typologie von Koordinationsformen; nach Weyer, 2000: 7, leicht verändert			
Koordinationsstyp	Markt	Hierarchie (Unternehmen/Staat)	Netzwerk
Koordinationsmittel	Preise	Formale Regeln, bürokratische Autorität	Vertrauen
Koordinationsform	spontan, spezifisch	geregelt, unspezifisch	diskursiv
Akteursbeziehungen	unabhängig, Verträge	abhängig, Mitgliedschaftsrolle	interdependent, Reziprozität
Zugang	offen	geregelt	begrenzt, exklusiv
Zeithorizont	kurzfristig	langfristig	mittelfristig
Konfliktregulierung	Recht	Macht	Verhandlung

Innerhalb der Diskussion über Netzwerke als eigenständige Organisationsform wurden weitere Merkmale für diese Eigenständigkeit hervorgehoben. Dabei wurden nicht nur die in der Tabelle genannten Merkmale Reziprozität und Interdependenz neben Vertrauen und Verhandlung genannt, sondern auch Reputation, Selbstverpflichtung/Selbstbindung, autonome Akteure, horizontale Verflechtung, gemeinsame Sprache/Kultur, Zielkongruenz, Moral, Mythos und Verlässlichkeit vorgeschlagen (vgl. Windeler 2001: 237ff.; Staber 2000; Powell 1996; Adler 2001; Weise 2000: 181ff.; Teubner 1996; Mayntz 1992; Jones et al. 1997; Mahnkopf 1994, Rölle, Blättel-Mink 1998; Sydow, Windeler 2000; Hirsch-Kreinsen 2002; Weyer 2000; Krömmelbein, Schmid 2000; Dolata 2003). Netzwerke werden auch informale Gemeinschaften genannt, wobei spontane Solidarität neben Vertrauen vorherrscht sowie binnenmoralische Prinzipien und soziale Normen gelten (vgl. Wiesenthal 2005: 231; Adler 2001: 219f.; Weyer 2000; Streeck, Schmitter 1985). Diese Merkmale sind aber kooperative Elemente, die je nach Autor u.U. als alleiniger Ausdruck einer Kooperation gelten sollen.

Gerade folgende Definition zeigt, dass Kooperation zur Abgrenzung der Organisationsform Netzwerk herangezogen wird.

„Ein Unternehmensnetzwerk stellt eine auf die Realisierung von Wettbewerbsvorteilen zielende Organisationsform ökonomischer Aktivitäten dar, die sich durch komplex-reziproke, eher kooperative denn kompetitive und relativ stabile Beziehungen zwischen rechtlich selbständigen, wirtschaftlich jedoch zumeist abhängigen Unternehmungen auszeichnet. Ein derartiges Netzwerk, das entweder in einer oder in mehreren miteinander verflochtenen Branchen agiert, ist das Ergebnis einer Unternehmungsgrenzen übergreifenden Differenzierung und Integration ökonomischer Aktivitäten“ (Sydow, 1992: 79; vgl. Sydow, Windeler 2004: 40).

Bei der Organisationsform Unternehmen/Staat wird oftmals von Hierarchie gesprochen. Die Organisationsform Markt bezieht sich auf Wettbewerb, wenn von dem Koordinationsinstrument Preis ausgegangen wird. Die Autoren führen die Eigenständigkeit der Organi-

<sup>26</sup> Ein Beispiel für eine Mischung von intermediärer und eigenständiger Form zeigt Niechoj (2003: 252).

sationsformen auf eine charakteristische Koordination zurück, die wiederum weitere eigenständige Merkmale beinhaltet, die sie von den anderen unterscheidet. Dabei wird die Auffassung vertreten, dass Markt mit Wettbewerb, Unternehmen mit Hierarchie und Netzwerk mit Kooperation gleichzusetzen ist (s.a. Becher et al. 2007).

Es zeigt sich bei genauer Betrachtung der Merkmale der Organisationsformen, dass eine Begründung der einzelnen Merkmale z.B. in Tab. 3-1 zwischen Koordinationsmittel, Koordinationsform und Konfliktregulierung nicht erfolgt. Verschiedene Merkmale der Netzwerke werden nicht zueinander in Beziehung gesetzt. Es bleibt bei einer „plausiblen Exemplifizierung“ und „Eigenschaftsheuristik“. Die Unterscheidungskriterien der Merkmale werden auch nicht dargelegt. Es wird nicht zwischen der Koordination und alleinigen Netzwerkmerkmalen oder zwischen „Koordinationsmechanismus im Sinne eines Prinzips der Handlungssteuerung und empirischer Koordinationsweise als Set praktischer Handlungsorientierungen“ unterschieden (vgl. Wiesenthal 2005: 232; vgl. Sydow, Windeler 2000: 11ff.).

Ein überwiegender Anteil der Autoren geht nicht explizit auf die Steuerung bzw. Koordination der Organisationsformen ein. Sie konzentrieren sich eher auf die Identifizierung der Netzwerkmerkmale. Einige Autoren untersuchen aber die Steuerung bzw. Koordination der Organisationsformen (Staber 2000; Osterloh, Weibel 2000; Sydow, Windeler 2000).

#### **3.1.4. Weitere Entwicklungen innerhalb Unternehmen, Märkten und Netzwerken**

##### **3.1.4.1. Empirischer Nachweis von Wettbewerb, Hierarchie und Kooperation in Organisationsformen**

Im theoretischen Diskurs wird, wie oben dargelegt, Markt mit Wettbewerb, Unternehmen/Staat mit Hierarchie und Netzwerk mit Kooperation gleichgesetzt. Diese Gleichsetzung ist aber nicht haltbar, da neuere Entwicklungen in bzw. Erkenntnisse über Organisationsformen auf ein gleichzeitiges Auftreten von Wettbewerb, Hierarchie und Kooperation als Koordinationsformen hindeuten. Diese Erkenntnisse im Zusammenhang mit dem gleichzeitigen Auftreten werden zuerst für Unternehmen, dann Markt und zum Schluss für Netzwerke dargestellt.

- Erstens hat die Organisationsform Unternehmen sich von einer stark hierarchischen Form über eine multidivisionale Form mit verschiedenen Unternehmensteilen zu einer eher netzwerkartigen Struktur entwickelt und sich dabei von dem Idealbild einer Hierarchie entfernt (Mahnkopf 1994: 75; Brand, 2002: 27ff., 101ff.; Jansen 2000: 20). Auch Dose (2005) diskutiert Begrenzungen und eine mögliche Flexibilisierung von bürokratischen Organisationen. Diese Entwicklung lässt sich auch als eine Veränderung oder Abschwächung der Hierarchie mit einer nachfolgenden Invasion bzw. Eingliederung von marktlichen oder netzwerkartigen Elementen in Unternehmen historisch interpretieren. Es gilt heute jedenfalls als gesichert, dass in Unternehmen neben einer Hierarchie auch marktliche und netzwerkartige Steuerungselemente auftreten. Beispielsweise thematisiert die sozialpsychologische Forschung schon lange Koordinationsformen in einer hierarchisch verregelten Organisation: Demokratische Führung als kooperative, autokratische Führung mit Lob und Tadel als hierarchische Koordination stehen neben der Laissez-faire-Führung, die nicht unbedingt als Wettbewerb identifiziert werden kann (vgl. Sader 1991: 272ff.; Rosenstiel et al. 1995). Die dominante Hierarchie scheint aber nicht zu verschwinden, sondern ihre Koordinationsmittel verändern sich und neue treten hinzu (vgl. Picot et al. 1996; Zenger 2002; Kawai, Schmid 2004; Moldaschl, Sauer 2000; Kühl 1999; Hirsch-Kreinsen 1995; Grunwald, Lilje 1981). Ein Beispiel für netzwerkartige Elemente in Unternehmen ist das Auftreten von Intraorganisationsnetzwerken wie Projektarbeit, Gruppenarbeit oder von Wissensnetzwerken in Unternehmen (vgl. Kühl 1999; Böhle, Bolte 2002; Kowol, Krohn

1997: 55; Renz 2003).<sup>27</sup> Marktliche Elemente im Unternehmen sind z.B. Leistungsentlohnungen mit Zielvereinbarungen als kooperatives Element gefolgt von Konkurrenz zwischen Abteilungen oder Gruppen (vgl. Moldaschl, Sauer 2000; Renz 2003; Voswinkel 2005; Jung 2005: 878ff.). Bei Finanzdienstleistern, wie einer Filiale einer Großbank, einem Call Center einer Direktbank, einer Bausparkasse, wurden neben der dominanten Hierarchie auch Wettbewerb zwischen Abteilungen/Arbeitsteams und eine begrenzte Kooperation in Form von Kollegialität zwischen den Arbeitsteams festgestellt (vgl. Dose 2005).

- Zweitens gab es auf dem Markt neben dem dominierenden Wettbewerb schon immer Kooperation zwischen den Marktseiten (vgl. Pies 2001; Bruegger 1999: 197ff.; Wehrich, Dunkel 2003; Jansen 2000: 19f.). Zusätzlich kreiert Wettbewerb Macht: Auf der einen Seite in dem ein Akteur eine größere Auswahlmöglichkeit hat als andere Akteure (Jansen 1999: 156). Auf der anderen Seite erlangt ein schwächerer Akteur durch das Eingehen einer Vertragsbeziehung (durch Investitionen) eine erhöhte Macht (s. fundamentale Transformation; Williamson 1985).

Werden Märkte als empirische Phänomene betrachtet, gibt es außer für den „Gesundheitsmarkt“ (Sachverständigenrat Gesundheitswesen 2005) und für den elektronischen Arbeitsmarkt (Becher et al. 2007; Kavai, Schmid 2004: 250ff.) keine systematischen Studien zum gleichzeitigen Auftreten von Wettbewerb, Hierarchie und Kooperation als Koordinationsformen bekannt. Der Sachverständigenrat Gesundheitswesen (2005: 48ff.) sieht als Koordinationsinstrumente erstens den Preismechanismus für Wettbewerb, die staatlich-administrative Steuerung als Hierarchie und korporative Kooperation mit Interaktionen zwischen Gesundheitsverbänden im Gesundheitsbereich. Für den elektronischen Arbeitsmarkt konnte z.B. nachgewiesen werden, dass neben vorherrschendem Wettbewerb auf dem externen elektronischen und vorherrschender Hierarchie auf dem internen elektronischen Teilarbeitsmarkt auch andere Koordinationsformen auftreten. Außerdem gibt es Arbeitsmarktbereiche, in denen Wettbewerb, Hierarchie oder Kooperation dominieren.<sup>28</sup>

- Drittens sind auch in Netzwerken Wettbewerb, Hierarchie und Kooperation vorhanden. Auf der einen Seite wird von Coopetition bzw. Konkurrenz mit gleichzeitiger Kooperation zwischen Unternehmen gesprochen, wobei Wettbewerb u.a. durch langfristige Interessensgemeinschaften und Eingebundenheit in eine Kultur bzw. in ein Umfeld „domestiziert“ wurde (vgl. Reiß 1998; Oughton, Whittam 1997; Sydow, Windeler 2000; Uzzi 1997; Corsten 2001; Powell 1996: 236). Auf der anderen Seite wird auch Macht bzw. hierarchische Beziehungen in Verbindung mit Vertrauen bzw. Kooperation in Netzwerken diskutiert (vgl. Bachmann 2000; Mahnkopf 1994; Windeler 2001: 39ff.; Hirsch-Kreinsen 2002). Gerade bei Projektnetzwerken mischen sich netzwerkförmige, hierarchisch-unternehmerische und marktliche Koordination (Windeler, Sydow 2004: 12f.). Zuliefernetzwerke werden überwiegend durch ein fokales Unternehmen beherrscht, aber auch kooperative Elemente spielen eine Rolle (z.B. in der Automobilindustrie durch einen großen Markenhersteller: Heidling 2000; vgl. Pohlmann 1996). Bei Innovationsnetzwerken oder virtuellen Unternehmen als Netzwerken wird sowohl von einer Gleichstellung ausgegangen als auch hierarchische Elemente gesehen (vgl. Windeler 2003; Brand 2004; Mahnkopf 1994: 76ff.). Auch in der politologischen Verhandlungstheorie wird von majoritären (als wettbewerblichen), hierarchischen und kooperativen Entscheidungssystemen gesprochen, die auch als Netz-

---

<sup>27</sup> Nach Böhle und Bolte (2002) gab es informelle Beziehungen mit kooperativem Charakter schon immer im Unternehmen, diese wurden im Taylorismus unterdrückt, aber in den letzten Jahren als produktive Ressource erkannt und als Projekt- bzw. Gruppenarbeit neben der bisherigen tayloristischen Arbeitsorganisation eingeführt.

<sup>28</sup> Historische Veränderungen wie bei Unternehmen wurden bisher nicht auf Märkten untersucht. Hier liegt weiterer Forschungsbedarf.

werke betrachtet werden können (vgl. Altemeier 1999: 31-72; Scharpf 1992). Systematische Untersuchungen über Wettbewerb, Hierarchie und Kooperation in Netzwerken bestehen i.W. nicht.

#### 3.1.4.2. *Theoretischer Diskurs als Koordinationsdiskurs – Vermischung des Organisations- und Koordinationsformenbegriffs*

Die genannten empirischen Befunde setzen sich erst langsam im theoretischen Diskurs durch. Dabei ist der Diskurs über Organisationsformen bzw. Netzwerke eher ein Koordinations-/Steuerungsdiskurs (vgl. Wiesenthal 2005; Müller-Jentsch 2003: 131ff.), was auch auf die ökonomischen Begründer der verschiedenen Diskursströmungen Coase (1937) und Williamson (1985) zurückzuführen ist. Beide Autoren trieb die Frage an, ob Markt oder Unternehmen (Hierarchie) die effizientere Organisationsform ist. Daraus entwickelte sich u.a. die politische Steuerungsdiskussion (vgl. Wiesenthal 2005), aber auch die soziologische mit Bezug auf Unternehmensnetzwerke (vgl. Müller-Jentsch 2003: 131ff.). Vermutlich spielt Steuerung eine Rolle im politologischen und soziologischen (u.U. auch ökonomischen) Netzwerkdiskurs, da angenommen wird, dass in Netzwerken (s. Anfang des Kap. 3.1.) z.B. mit den sich häufig ändernden, krisenhaften sozio-ökonomischen Entwicklungstendenzen am besten umgegangen werden kann. Es wird nach den Ansatzpunkten und Möglichkeiten der Steuerung und Anpassung der erfolgreichen Netzwerke, aber auch u.a. von Märkten oder bürokratischen Organisationen, bei einer sich ständig ändernden Umwelt gesucht. Der Steuerungsdiskurs, der sich nur um Steuerung bzw. Koordination dreht, stellt aber nur einen kleinen Ausschnitt des Gesamtdiskurses dar. Erkenntnisse aus diesem Diskurs gehen aber in die Diskussion und Definition über die Steuerung/Koordination in Kapitel 3.2.3. ein.

Nur vereinzelte Autoren vertreten die Auffassung, dass Koordinations-/Organisationsformen in verschiedenen Mischungen auftreten und Koordinationsformen in Organisationsformen vorkommen.<sup>29</sup> Bradach/Eccles (1991), Becher et al. (2007) und Wiesenthal (2005) zählen zu diesen wenigen Autoren. Sie sind der Meinung, dass sich Markt, Gemeinschaft/Netzwerk und Unternehmen in Realität nicht ausschließen, sondern in verschiedenen Mischformen (Bradach/Eccles: „plural forms“) auftreten. Sie fassen Markt, Gemeinschaft/Netzwerk und Unternehmen als Idealtypen auf, die in Wirklichkeit überlappend, ineinander verwoben oder komplementär sein können. Wiesenthal stellt sich dies als Metapher von Schiebereglern mit verschiedener Relevanz bei Markt, Gemeinschaft/Netzwerk und Unternehmen vor. Becher et al. (2007) argumentieren ähnlich und versuchen die Koordinationsformen durch die Transaktionskostentheorie innerhalb des externen und internen elektronischen Arbeitsmarkts zu erklären.<sup>30</sup>

Zusätzlich wird zwischen den Organisationsformen und ihrer Koordination nicht deutlich unterschieden. Wiesenthal (2005: 232) schreibt: „Alle beobachtbaren Sozialformen schei-

---

<sup>29</sup> Kieser und Kubicek (1992: 104f., 106f., 118) sehen zwar die Koordinationsinstrumente Weisung als dominierendes hierarchisches Element und Selbstabstimmung bzw. Unternehmenskultur als kooperative Elemente in Unternehmen. Wettbewerbliche Strukturen durch Profitcenters werden zwar auch benannt, aber eine systematische Untersuchung und Bedeutungsabschätzung erfolgt nicht.

<sup>30</sup> Interessanterweise treten die Koordinationsformen Hierarchie/Macht, Kooperation und Wettbewerb bei Müller-Jentsch (2003: 134ff.) auf verschiedenen Ebenen auf. Macht und Vertrauen sind Steuerungsmedien, die einen Bezug zur gesellschaftlichen Ebene haben. Auf einer weniger abstrakten Ebene bestehen Regeln als Steuerungsinstrumente, wie Kooperation, Aushandlung und Wettbewerb. Begründet wird nicht, warum die verschiedenen Koordinationsformen auf verschiedenen Ebenen liegen. Die Beziehung der kooperativen Elemente wie Vertrauen, Aushandlung und auch Reziprozität zur Kooperation wird nicht thematisiert. Eine dezidierte Auseinandersetzung mit dem Kooperationsbegriff erfolgte anscheinend wegen eines Fokus auf die Netzwerksteuerung nicht.

nen unter dem Dach des jeweils ‚führenden‘ Prinzips auch die übrigen Prinzipien zu beheimaten.“ Dabei unterscheidet er zwischen dem Koordinationsmechanismus als Prinzip der Handlungssteuerung und der empirischen Koordinationsweise als „Set praktischer Handlungsorientierungen“. Beim Sachverständigenrat für Gesundheit (2005: 50) in seiner Studie zum Gesundheitswesen ist dies zwar nicht ganz so deutlich, doch wird Wettbewerb nicht nur im Markt, sondern auch bei korporatistischer Kooperation und bei staatlich-administrativer Steuerung gesehen. Der Markt ist ein Koordinationsmechanismus zur Abstimmung sozialer Interaktionen, als auch eine konkrete Organisation dieser Abstimmungen. Einzig Becher et al. (2007) unterscheiden deutlich zwischen den Koordinationsformen Wettbewerb, Kooperation und Macht, die in den Organisationsformen Markt, Unternehmen und Netzwerk auftreten. Dabei können die Organisationsformen jeweils unterschiedliche Ausprägungen bzw. Koordinationsmittel/-mechanismen beinhalten.

### **3.2. Koordination und Organisation als Institutionen – Weiterführende Überlegungen und Neudefinition**

Ausgehend von obigen Diskursen um Netzwerk, Koordination und Organisationsformen erfolgt im Folgenden eine bisher nicht erbrachte, systematische Auseinandersetzung mit Koordination, Koordinationsformen, Organisation und Organisationsformen. Da Mischungen der Koordinationsformen Hierarchie, Kooperation und Wettbewerb in Organisationsformen auftreten, wird zwischen Organisationsform und Koordinationsform unterschieden (vgl. Becher et al. 2007). Eine Gleichsetzung von (1) Netzwerk mit der Koordination durch Kooperation, (2) Unternehmen/Staat mit Hierarchie und (3) Markt mit Wettbewerb wird fallengelassen. Zusätzlich wird später das Arbeitshandeln beim Softwareentwicklungsprozess in der Organisationsform Netzwerk in Form eines Arbeitsnetzes untersucht, so dass der Fokus nicht auf der Organisationsform, sondern auf Koordinationsformen auf der Handlungsebene liegt.

Ein weiterer Ausgangspunkt einer Auseinandersetzung mit Koordinations- und Organisationsformen ist die Erkenntnis, dass diese als Institutionen verstanden werden können. Daher wird zunächst der Institutionenbegriff betrachtet, der Institutionen in Verbindung mit einer Leitidee sieht (s. Kap. 3.2.1.). Ausgehend vom Institutionenbegriff werden Organisation, Organisationsformen auf einer organisationellen Ebene (s. Kap. 3.2.2.) und Koordination, Koordinationsformen auf einer darunterliegenden Handlungsebene (s. Kap. 3.2.3.) anhand von Leitideen definiert. Koordinationsformen sind spezifische Institutionen der Koordination, die sich wiederum aus verschiedenen Regeln zusammensetzen.

Eine Handlung erfolgt einerseits auf der Ebene der Individuen. Einzelne normorientierte Handlungen können den Koordinationsformen zugeordnet werden. Andererseits werden hier Arbeitsprozesse zwischen individueller- und organisationeller Ebene untersucht, in denen mehrere Handlungen zu Teilprozessen zusammengefasst werden. Koordinationsformen treten bei Handlungen der Akteure als Verhaltensnormen auf und werden als solche untersucht, können aber auch als Werte in der gesellschaftlichen Makroebene vorkommen (s. z.B. Cultural Theory: Thompson et al. 1990), die hier aber nicht betrachtet werden. Die Definition der drei Koordinationsformen erfolgt dabei anhand eines Kategorisierungsschemas, dessen Kategorisierungsdimensionen, anders als in den Diskursen, in Beziehung zueinander stehen. Die Genese der Koordinationsformen bleibt ausgeblendet, jedoch wird die Regel als Zustand definiert. Als Koordinationsformen werden Wettbewerb, Kooperation und Hierarchie/Macht, als Organisationsformen Markt, bürokratische Organisation (z.B. Unternehmen, staatliche Organisationen oder Non-Profit-Verbände) und Netzwerk bestimmt.

### **3.2.1. Definition des Institutionenbegriffs**

Der Begriff Institution geht auf Durkheim (1984) zurück, der diese als „Glaubensvorstellungen und durch die Gesellschaft festgesetzte ... Verhaltensweisen“ bezeichnete. Eine allgemeine Definition des Institutionenbegriffs besteht i.W. nicht, es wird jedoch u.a. in Lexikonartikeln die Stabilität und Dauerhaftigkeit von Handlungs- bzw. Beziehungsmustern hervorgehoben (s. Hillmann 1994: 375; Fuchs-Heinritz, et al. 1995: 302). Im soziologischen Neo-Institutionalismus werden Institutionen als Erwartungsstrukturen konzeptionalisiert (vgl. Hasse, Krücken 1999; Senge 2005: 20, 109ff.; Türk 2000). In der Neuen Institutionenökonomie wird unter Institutionen ein System von Regeln verstanden, das Ordnung in das menschliche Verhalten bringt (Richter, Furubotn 1996: 7ff.; Göbel 2002: 1f.).

Als vorläufige Definition können Institutionen als sinnhaftes, stabiles Handeln verstanden werden, wobei sich die Individuen an Erwartungen anderer ausrichten. Das Stabile beim Handeln sind die Regeln, die in der Praxis immer wieder reproduziert werden (vgl. Giddens 1988: 68ff.).

Institutionen schränken zwar auf der einen Seite das Verhalten ein, auf der anderen Seite ermöglichen sie eine soziale Ordnung der Verhaltensweisen bzw. ein geordnetes Miteinander. Feststehende Verhaltensweisen der sozialen Ordnung sind keine naturwissenschaftlichen Gesetze bzw. fest einprogrammierte Verfahrensabläufe, sondern sie unterliegen durch die Reproduktion in der Praxis einer Variation, die neue Institutionen bzw. eine neue soziale Ordnung gebiert. Der einschränkende und ordnungsgebende Charakter der Institutionen ist die stabile und die Verhaltensvariationen die dynamische Seite der Institution. Institutionen werden also entweder als dem Individuum von außen gegenüber tretende Struktur, d.h. als objektiver Zustand, oder als ein dynamisches System verstanden, das sich an neue Gegebenheiten anpasst bzw. angepasst wird (vgl. Durkheim 1984; Walgenbach 1999: 320f.; DiMaggio, Powell 1991b: 9; Scott 2000: 50; Schüle 1987; Giddens 1988: 67ff.). Die Erhaltung sowie die Veränderung oder Entstehung von Institutionen bzw. Regeln ist nicht unabhängig vom Handeln, sondern liegt innerhalb der Praxis der sozialen Beziehungen. Die Regeln werden also während des Handelns reproduziert oder verändert (Giddens 1988: 78f.). Institutionen lassen sich dennoch analytisch aus der Perspektive ihrer Institutionalisierung, d.h. als einen Prozess der Entstehung und Auflösung oder als einen Zustand betrachten.

Zur Vereinfachung und als erste Annäherung wird in dieser Arbeit die Perspektive des objektivierte Zustands der Institutionen und nicht des Prozesses des Institutionenwandels eingenommen. Damit werden die Regeln als statisch betrachtet und eine dynamische Perspektive ausgeschlossen.

Eine Institution als Regelsystem wird dabei durch eine bestimmte gemeinsame Leitidee bestimmt, die die Regeln zusammenhält. Die Institution Ehe beispielsweise fasst die Regeln Hochzeit, gemeinsame Erziehung von Kindern etc. zusammen (Gukenbiehl 1993: 98, 102). Eine Institution ist also ein wiederholt auftretendes Verhaltensmuster, das aus einem Regelsystem besteht, das durch Erwartungen um eine gemeinsame Leitidee generiert wird. Regeln sind wiederholt auftretende Verhaltensmuster auf der Akteurs- bzw. Handlungsebene (vgl. Gukenbiehl 1993). Auf der Handlungsebene sind also Regeln und Normen Verhaltens-, Verfahrens-, „anweisungen“ für spezifische, zeitlich abgrenzbare, punktuelle soziale Situationen. Institutionen fassen die Regeln der Handlungsebene durch allgemeine Leitideen zusammen. Dabei unterscheiden sich Institutionen durch den Umfang bzw. Anzahl der Normen bzw. Regeln, die sie umfassen. Die Leitidee wird abstrakter, je mehr Normen die Institution umfasst. Auf einer individuellen Ebene ist die Leitidee weniger abstrakt als auf einer Meso- oder Makroebene. Anhand der Leitidee können also die nachfolgenden Institutionen „Organisation“, „Organisationsform“, „Koordination“ und „Koordina-

tionsform“ auf verschiedenen Ebenen definiert werden, wobei sich der Abstraktionsgrad der Leitidee mit der jeweiligen Ebene unterscheidet.

#### **3.2.2. Definition der Organisationsformen**

Die folgende Definition der Organisation abgeleitet vom Institutionenbegriff mit der zentralen Leitidee ist weitgehend üblich und recht einfach gehalten. Dagegen ist die Definition der einzelnen Organisationsformen Markt, bürokratisch-rationale Organisation (wie Unternehmen) und Netzwerk recht komplex, da einerseits der Begriff Organisation auch für bürokratische Organisationen verwendet wird, andererseits die Organisationsformen mit den Koordinationsformen begrifflich verbunden sind (s.u.).

##### *3.2.2.1. Organisationsform als Institution*

Eine Organisation ist ein Beispiel einer Institution auf der Mesoebene. Der Begriff Organisation hat aber eine doppelte Konnotation. Einerseits wird darunter eine Organisationsform mit hierarchischen Strukturen verstanden (s.o.). Andererseits bezeichnet der Begriff alle Organisationsformen und stellt damit eine einzige, übergreifende „Meta“-Organisationsform dar. An dieser Stelle werden unter dem Begriff Organisation alle Organisationsformen verstanden, die auf der organisationellen Mesoebene zwischen Mikro- und gesellschaftlicher Makroebene wiederzufinden sind und verschiedene Regeln beinhalten. Sie sind abgrenzbare, eigenständige, ziel- und zweckgerichtete Entitäten, die künstlich errichtet wurden und in denen eine Ordnung durch Arbeitsteilung herrscht. Die Organisationen haben als Institutionen eine künstlich geschaffene institutionelle Leitidee, die mit der organisationalen Zielorientierung zusammenhängt und damit die Regeln in einer Organisation bündelt bzw. zusammenführt. Sie wird üblicherweise bei der Gründung einer Organisation festgelegt (vgl. Gukenbiehl 1993: 104f.; Richter, Furubotn 1996: 8f.; Hillmann 1994: 638; Fuchs-Heinritz, et al. 1995: 478; Kieser, Kubicek 1992: 95). Das Organisationsziel als Leitidee stellt somit die Basis und Lebensgrundlage der Organisation dar, weswegen sie sehr widerstandsfähig gegenüber gravierenden Änderungen scheint.

Da Leitideen „nicht lediglich beliebig und mehr oder weniger folgenlos in den Köpfen von Individuen herrschen sollen, muss sich die Leitidee in den Rationalitätskriterien, Zielen, Strukturen und Rollen einer Organisation manifestieren, darüber hinaus aber auch in den alltäglichen Selbst-Definitionen, Beobachtungsweisen und Entscheidungen der Akteure in der Organisation auffinden lassen.“ (Edeling 2002: 228).

Eine Leitidee durchdringt damit eine Organisationsform, die sich anhand unterschiedlicher Ausformung einer übergreifenden Leitidee unterscheiden lassen. Die Leitidee muss dabei zum einen in allen Organisationsformen zu finden sein, da diese alle zu den Organisationen gehören. Zum anderen muss die Leitidee spezifizierbar für eine Organisationsform sein. Gemeinsam ist allen Organisationsformen, dass alle durch eine Arbeitsteilung als übergreifende Leitidee gekennzeichnet sind (s.o.). Die Organisationsformen unterscheiden sich aber nach ihren Zielorientierungen bzw. spezifischen Leitideen, die jeweils eine andere Art des Umgangs mit der Arbeitsteilung hervorbringen.

Nach ihren Leitideen z.B. bzgl. der Arbeitsteilung können die drei idealtypischen Organisationsformen Markt, bürokratische Organisation (wie Unternehmen) und Netzwerke unterschieden werden. Weiterhin bestehen spezifische (Unter-)Formen der Organisationsformen, die sich wiederum leicht in ihren Zielorientierungen bzgl. allgemeinen und spezifischen Zielen unterscheiden. Die jeweilige Leitidee dieser drei Organisationsformen ist ein zentrales Merkmal, nach dem die jeweilige Organisationsform definiert werden kann. Dieses Definitionsmerkmal sollte aber einerseits eine eigenständige Organisationsform von den anderen abgrenzen. Andererseits sollte dieses Merkmal nicht mit der Koordination

und den Koordinationsformen zusammenhängen, da eine begriffliche Trennung zwischen Organisations- und Koordinationsformen angestrebt wird.

Da diese drei Organisationsformen zwischen dem Gesellschaftssystem auf der Makroebene und der Koordinationsform auf einer Handlungsebene (Mikroebene) stehen, könnte deren Leitidee bzgl. der Arbeitsteilung einen Bezug zu einem spezifischen Gesellschaftssystem aufweisen und damit ohne Koordinationsbezug definiert werden. Die symbolisch generalisierten Kommunikationsmedien von Gesellschaftssystemen nach Luhmann oder Parsons könnten eine Repräsentanz als Leitideen in den Organisationsformen finden, z.B. Geld, Macht und Vertrauen als Steuerungsmedien nach Müller-Jentsch (2003: 134f.). Doch können die Organisationsformen mit ihren Steuerungsmedien nicht eindeutig einem Gesellschaftssystem zugeordnet werden. Nach Müller-Jentsch tritt Geld als Steuerungsmedium des Wirtschaftssystems in allen drei Organisationsformen auf, wobei Netzwerk als Unternehmensnetzwerk verstanden wird. Das Steuerungsmedium Macht gehört zum politischen System, tritt aber nicht nur dort auf. Vertrauen kann (als Systemvertrauen) keinem Gesellschaftssystem zugeordnet werden. Außerdem sind Macht und Vertrauen in allen drei Organisationsformen vorhanden (s.o.). Keine der genannten Steuerungsmedien aus Gesellschaftssystemen kann daher als Leitidee fungieren.

Da die Leitideen nicht durch die Makroebene erklärbar sind, wird der Blick auf den Zusammenhang zwischen Organisationsformen und Handlungsebene gerichtet. Spezifische Regeln aus der Handlungsebene werden in einer Organisationsform gebündelt. Es besteht eine andere Sichtweise: die Organisationsformen legen nicht ihre Leitideen fest bzw. sind von vorneherein bestimmt, sondern die Akteure entwickeln die Leitidee auch bzgl. der Arbeitsteilung, die einer Organisationsform zugrundeliegt. Nach Hollingsworth und Boyer (1997), aber auch Wiesenthal, fungieren einzelne Koordinationsmechanismen als „Vorstellungshorizonte“, nach denen Akteure ihre Erwartungen, Bedürfnisse und Vorlieben ausrichten.

„Alles in allem gibt die Empirie wenig Grund zur Annahme, bei den drei basalen Mechanismen [Markt, Unternehmen und Netzwerk, d.A.] handele es sich um je für sich koordinationsstaugliche Regelungsmechanismen. Vielmehr scheint es, als sei das Reden über distinkte Koordinationsprinzipien einem anderen Ausschnitt der sozialen Wirklichkeit zuzurechnen als die Praxis sozialer Interaktion. Kategorial distinkte Koordinationsmechanismen wären danach lediglich soziale Konstruktionen, die sich deshalb so gut zum Aufbau von Interaktionssystemen [d.h. Organisationsformen, d.A.] eignen, weil sie den Akteuren prononcierte Vokabularien bieten“ (Wiesenthal 2005: 235).

Eine Organisationsform als Koordinationsform tritt also nicht als reale Handlung auf, sondern ist eine Orientierungshilfe für die Reflektion. Damit kann auch eine Koordinationsform bzw. ein –mechanismus mit Regelbündeln als eine Leitidee auftreten. Die spezifisch zu untersuchende Leitidee einer Organisationsform z.B. bzgl. der Arbeitsteilung ist dabei ein soziales Konstrukt nach dem Organisationsformen und Koordinationsformen einerseits differenziert werden können. Andererseits besteht anscheinend eine vollständige Trennung zwischen Organisations- und Koordinationsform nicht, was an der sozialen Konstruktion liegen kann.

#### 3.2.2.2. *Definitionen von Markt, bürokratischer Organisation und Netzwerk*

Im Folgenden werden die Organisationsformen als Idealtypen definiert, wobei die zentrale Leitidee einer Arbeitsteilung für Markt, bürokratische Organisation und Netzwerk spezifiziert wird. Dies erfolgt, um die Organisationsformen von den Koordinationsformen abgrenzen zu können. Auch da, wie oben argumentiert wurde, Organisationsformen und Koordinationsformen eher soziale Konstrukte bzw. Abstraktionen sind. Dabei wird die Dominanz einer Koordination impliziert, wie im folgenden dargelegt wird.

Basis der Definitionen ist, angelehnt an die formale Netzwerkanalyse, das bestehende Beziehungsnetzwerk und individuelle oder institutionelle Akteure. Die zentrale Leitidee einer Organisationsform beeinflusst dabei die unterschiedlichen sozialen, vernetzten Beziehungen zwischen den Akteuren in einer durch multivernetzte und multikomplexe Beziehungen bestimmten Welt. Die Leitidee spielt in der inhaltlichen Beziehungsdimension zwischen den Akteuren eine Rolle. Ob sie Nicht-Beziehungen als Grenzen zwischen den Organisationsformen entstehen lässt, lässt sich in dieser Arbeit nicht sagen.

Die Definition der drei Organisationsformen erfolgt anhand der Literatur, da damit einerseits die Koordinationsaspekte weitestgehend ausgeblendet, andererseits die diskutierten Merkmale nach ihrem Bezug zur Arbeitsteilung untersucht werden können. Die Koordinationsformen können aber nicht vollständig ausgeblendet werden, wie obige Argumentation zeigt, so dass spezifische Ausprägungen der Koordinationsformen in den Definitionen der Organisationsformen aufscheinen.

Die Leitidee der drei idealtypischen Organisationsformen Markt, bürokratische Organisation und Netzwerk sind jeweils spezifische Formen der Arbeitsteilung, die im Folgenden genannt werden. Weitere Strukturmerkmale wie weitere Beziehungsdimensionen und Merkmale von Akteuren werden herangezogen, um die Organisationsformdefinitionen anzureichern. Unter Akteursmerkmalen werden individuelle wie auch korporative Akteure verstanden, da dies die zentrale Leitidee für eine Definition nicht beeinflusst. Individuen sind dabei generell Träger der Beziehungsnetzwerke, die Organisationsformen unterscheiden sich aber nach dem Beziehungsinhalt. So können Individuen auch als Vertreter einer bürokratischen Organisation (wie Unternehmen oder Verband) als spezifische Beziehungsdimension auftreten. Zunächst wird die Organisationsform Markt, dann bürokratische Organisation und zum Schluss Netzwerk anhand zentraler Leitidee und weiteren Strukturmerkmalen unterschieden und definiert.

- Bei Märkten als Organisationsformen steht eine atomistische Arbeitsteilung durch Tausch im Vordergrund, d.h. Akteure nutzen Märkte mit dem Ziel des gleichgestellten Austauschs von Verfügungsrechten an Produkten und Dienstleistungen, die sie als Arbeitspakete erstellen bzw. erstellt haben. Dabei wird Angebot und Nachfrage über Preise, aber auch Quantität und Qualität des Produkts/der Dienstleistung abgeglichen (vgl. Pirker 2000: 72; Kieser, Kubicek 1992: 118). Die Akteure als Tauschpartner sind rechtlich und wirtschaftlich autonom und können Individuen wie auch bürokratische Organisationen (s.u., z.B. Unternehmen) sein. Der Austausch erfolgt zu einem spezifischen Zeitpunkt (vgl. Powell 1996: 223). Beispiele für die Organisationsform Markt sind Produkt-, Dienstleistungs- und Arbeitsmärkte.

- Bürokratische, rationale Organisationen, die auch unter dem Begriff Organisationen verstanden werden (s.o.), wurden mit dem Ziel der bürokratischen Verwaltung bzw. Standardisierung von Produktion und Dienstleistungserstellung künstlich geschaffen. Nach Max Weber (1980) ist diese Organisationsform nach rationalen Gesichtspunkten organisiert, besonders wenn es sich um eine Wirtschaftsorganisation handelt. Das jeweilige Organisationsziel soll also mittels des Rationalitätsprinzips erreicht werden. Damit erfolgt eine abhängige Arbeitsteilung durch formale, rationale Festlegung von Routinen durch Vorgesetzte. Ein weiteres idealtypisches Merkmal sind unbefristete, formale Beziehungen, die sich in Form einer unbefristeten Anstellung darstellen und u.a. mit einer Abhängigkeit der Akteure zusammenhängen. In Unternehmen treten vor allem Individuen, aber in Holdings auch (kleinere) Unternehmen als Akteure auf. Die formalen Abhängigkeitsbeziehungen werden über vom Unternehmer abgeleitete Verfügungsrechte begründet. Dabei wird in den bürokratischen Verbänden formales, explizites Wissen<sup>31</sup> generiert und über formale

kratischen Verbänden formales, explizites Wissen<sup>31</sup> generiert und über formale Kanäle zwischen Abteilungen ausgetauscht (vgl. Pirker 2000: 73). Unterarten sind Unternehmen, die als Ziel die Gewinnmaximierung bzw. gewinnorientierte Produktion haben, und staatliche Organisationen, deren Ziel die Verwaltung bzw. der Erhalt von öffentlichen Gütern ist (vgl. Schimank 2005: 24ff.; Voswinkel 2005: 287f.).

- Netzwerke hängen mit Basisnetzwerken („natürlichen“ Unterstützungsnetzwerken) zusammen, die Organisationsform Netzwerk unterscheidet sich von diesen durch ihre künstliche Natur. Die Organisationsform Netzwerk steht dabei in dieser Arbeit im Mittelpunkt, da Open Source-Projekte Arbeitsnetze darstellen.<sup>32</sup>

Die künstlichen Netzwerke ähneln stark freiwilligen, informalen und selbstorganisierten Gruppen bzw. Zusammenschlüssen, wobei auch Beziehungen über formale organisationale Grenzen hinweg bestehen können. Das Ziel eines Netzwerks hinsichtlich der selbstorganisierten Arbeitsteilung ist die informale und selbstorganisierte Realisierung von gemeinsamen Vorteilen, z.B. dem gemeinsamen Besitz eines (Dienstleistungs-)Endprodukts oder eines Ressourcenpools in Form einer Allmende (Clubgut, Allmendegut oder u.U. öffentliches Gut; vgl. Ostrom 1990; Müller-Jentsch 2003: 136). Dabei bestehen zwischen den Mitgliedern kurz- bis mittelfristige, z.T. wiederholte, eher informale Beziehungen, wobei auch implizites Wissen weitergegeben werden kann. Die Elemente dieses Idealtypus Netzwerk sind zum einen autonome Akteure, die auch auf dem Markt auftreten. Zum anderen ist die Kollaboration mit begrenztem, exklusivem Zugang ein weiteres Element des Netzwerks, das nach der Literatur vor allem zu Hierarchie/Unternehmen gehört. Die Akteure sind rechtlich, aber nicht unbedingt wirtschaftlich autonom. Die Netzwerke sind temporär und zugleich in ihrer Struktur stabil, d.h. die Akteure ändern sich, aber die grundlegenden Beziehungen bleiben gleich. Wie in Märkten oder Unternehmen treten in Netzwerken Individuen oder bürokratische Organisationen (z.B. Unternehmen) als Akteure auf. Individuelle oder korporative, bürokratische Akteure treten in Netzwerken wie Unternehmensnetzwerken von großen und kleinen Unternehmen, Projektnetzwerken, Arbeitsnetzen (Open Source-Projekt; s. Kap. 2), virtuellen Gruppen etc. auf. In Arbeitsnetzen als Unterform der Organisationsform Netzwerke wird als Vorteil ein gemeinsames Produkt durch gemeinsames Arbeiten angestrebt.

#### **3.2.3. Definition der Koordinationsformen**

In den obigen Definitionen der Organisationsformen wird eine Koordination implizit mitgedacht. Wie oben angedeutet, können die Organisationsformen kaum auf sich ausschließende Merkmale zurückgeführt werden, ohne dass die auf der Arbeitsteilung basierende Unterscheidung zwischen wettbewerblichem Gegeneinander und Miteinander der Zusammenarbeit zu erkennen ist (s.u.). Bei der bürokratischen Organisation können ohne weiteres hierarchische, bei Netzwerken kooperative und beim Markt wettbewerbliche Elemente eingepasst werden.

Dies hängt mit der von allen Organisationsformen geteilten Leitidee der Arbeitsteilung zusammen, wobei eine Arbeitsteilung nicht ohne eine Koordination verschiedener Individuen zu denken ist (Kieser, Kubicek 1992: 95; Laux, Liermann 2003: 5).<sup>33</sup> Außerdem unter-

---

<sup>31</sup> Zur genaueren Darstellung der Beziehung zwischen explizitem und implizitem Wissen s. Brödner 2007.

<sup>32</sup> Weitere Netzwerke sind z.B. Unternehmensnetzwerke, die nach Müller-Jentsch (2003: 135) eine sehr voraussetzungsvolle Organisationsform sind. Ein Unternehmensnetzwerk ist in einer Definition eine Neukombination von Elementen aus Markt und Unternehmen, wodurch eine eigenständige Organisationsform entsteht. Wirtschaftliche Interessen treten neben Hierarchie/Macht auf (vgl. Teubner 1996).

<sup>33</sup> Eine Arbeitsteilung impliziert ja gerade, dass verschiedene Individuen sich bei der Erstellung von Arbeitspaketen gegenseitig abstimmen.

scheiden sich die Organisationsformen in ihrer Ausformung der Arbeitsteilung. Diese spezifische Arbeitsteilung in einer Organisationsform kann dazu führen, dass eine dominierende Koordinationsform wahrgenommen wird, die besonders gut zur spezifischen Arbeitsteilung passt.<sup>34</sup>

Da die Perspektive aber nicht auf der Organisationsform liegt, sondern auf der Softwareentwicklung als Arbeit, ist davon auszugehen, dass zwar wahrscheinlich eine Koordinationsform dominiert, aber auch ein Blick auf die anderen Koordinationsformen angestrengt werden muss. Es kann nämlich sein, dass an bestimmten Punkten oder in Teilprozessen der Softwareentwicklung andere Koordinationsformen dominieren.

Die Koordination und die Koordinationsformen werden in dieser Untersuchung als Institutionen aufgefasst und daher mittels des Institutionenbegriffs definiert, um die Koordination und ihre Koordinationsformen zu fokussieren. Da der Diskurs über die Steuerung/Koordination mit dem Diskurs über die Organisationsformen Markt, bürokratische Organisation (Unternehmen/Staat/Verbände) und Netzwerk verbunden ist, wird dieser Diskurs nach brauchbaren und wichtigen Definitionen für die Koordination untersucht (s. Kap. 3.2.3.1.). Es wird nach einer zentralen Leitidee der Koordination als Institution gesucht, die charakteristisch für die Koordination auf der individuellen Ebene ist. Der Diskurs über Steuerung/Koordination (auch Regulation) stellt aber nur einen kleinen Ausschnitt des Diskurses im Zusammenhang mit Organisationsformen dar.

Im Anschluss werden die Koordinationsformen zuerst aus einer allgemeinen Begrifflichkeit des Gegeneinanders-Miteinanders definiert (s. Kap. 3.2.3.2.). Es werden Erkenntnisse aus Sozialpsychologie, Soziologie und Ökonomie zu den jeweiligen Koordinationsformen zusammengetragen, da Kooperation, Hierarchie/Macht und Wettbewerb grundlegende Handlungsregeln bzw. Verhaltensweisen darstellen, die als einzelne Phänomene in diesen Disziplinen intensiv untersucht wurden. Diese Erkenntnisse werden dann für Kooperation, Hierarchie/Macht und Wettbewerb anhand eines Kategorisierungsschemas im Hinblick auf Handlungsbedingung, Handlung und Handlungsergebnis verortet (s. Kap. 3.2.3.4., 3.2.3.5., 3.2.3.6., 3.2.3.7.), um der eklektischen Zusammenstellung der Kategorien bei den Organisationsformen zu entgehen (s.o.).

#### 3.2.3.1. *Definition der Koordination*

Organisationsform und Koordinationsform sind Institutionen auf verschiedenen (Abstraktions-)Ebenen eines Institutionengefüges, die in der bisherigen Diskussion nicht unterschieden wurden (s.o.). Sydow und Windeler (2000: 3ff.) sehen zwar in der Diskussion über die Steuerung von Unternehmensnetzwerken diese Steuerung auf verschiedenen Steuerungsebenen (wie Netzwerk, Organisation, Individuum und institutioneller Kontext) ablaufen, wobei durch den Einsatz von Kooperation, Vertrauen, Selbstverpflichtung, Verlässlichkeit, Verhandlung, dauerhafter Beziehungszusammenhang gesteuert wird.<sup>35</sup> Dahinter steht die Frage, auf welcher Ebene eine Steuerung ansetzen soll. Inwiefern diese Unterteilung in Steuerungsebenen sinnvoll ist, kann an dieser Stelle nicht geklärt werden.

In dieser Arbeit liegt der Fokus nicht auf der Organisation, die auf einer Mesoebene angesiedelt ist, sondern auf der Koordination einer darunterliegenden individuellen Handlungsebene. Auf dieser Handlungsebene verlaufen auch die Beziehungsnetzwerke zwischen

---

<sup>34</sup> Dieser Zusammenhang zwischen Koordination und Arbeitsteilung führt anscheinend dazu, dass die Organisationsformen in der Literatur bisher immer mit ihrer dominierenden Koordinationsform gedacht wurden (s. Kap. 3.1.).

<sup>35</sup> Müller-Jentsch (2003: 133ff.) sieht noch auf verschiedenen Ebenen entweder Macht oder Wettbewerb neben kooperativen Elementen als Steuerungsmechanismen.

den Akteuren, wobei die Handlungen mit den Beziehungen in Verbindung stehen. Da es sich bei den Organisationsformen um eine spezifische Arbeitsteilung als Leitidee dreht und Arbeitsteilung eine Koordination nötig macht, werden die Organisationsformen mit einer Koordination verbunden. Arbeitsteilung impliziert eine arbeitsbezogene Abhängigkeit, wobei Mitglieder einer Organisation auf das jeweilige Organisationsziel ausgerichtet werden (vgl. Kieser, Kubicek: 95; Göbel 2002). Die Frage nach der Steuerungsebene ist aber nicht weiter relevant, da eine Koordination einer Handlungsebene in dieser Arbeit untersucht wird. Im Weiteren wird die Leitidee der Koordination gesucht, d.h. der Modus bzw. das „Wie“ einer Koordination/Steuerung mit Ausrichtung auf das Organisationsziel.

Die Koordination wurde in der angloamerikanischen Literatur im Zusammenhang mit der Neuen Institutionenökonomie als „Governance“ thematisiert (vgl. Williamson 1985; Göbel 2002). Der Begriff „Governance“ bezeichnet auf der einen Seite eine soziale Kontrolle über einen Staat, eine Organisation, etc., aber auf der anderen Seite das Beeinflussen bzw. Festlegen von Aktivitäten (vgl. Thompson 1998). Diese beiden Verständnisse der Koordination, einerseits die externe Steuerung, andererseits das Abstimmen bzw. gegenseitige Festlegen, sind auch in der Literatur zur Steuerung hinsichtlich des Diskurses über Organisationsformen zu finden.

„Koordination ist die Zuordnung verschiedener Teilsysteme zu einer funktionierenden Gesamtheit. Koordinieren ist definiert als ordnendes Zusammenfassen, etwas aufeinander abstimmen, etwas in ein Gefüge einbauen, etwas miteinander in Einklang bringen. Die beteiligten Teilsysteme spielen bei dieser Zuordnung tendenziell eine passive Rolle. Hier setzt eine Unterscheidung zur Kooperation an: Bei kooperativen Systemen ordnen sich die beteiligten Systeme aktiv in eine Gesamtheit ein. Kontrolle ist im Falle der Koordination extern verortet, im Falle einer Kooperation bei den Handelnden selbst.“ (Kumbruck 2001: 159).

Diese definitorische Unterscheidung führt jedoch in die Irre. Implizit wird zwischen einem Gesamtsystem<sup>36</sup>, das die Subsysteme steuert, und den abzustimmenden Subsystemen bzw. Individuen unterschieden. Die Definition der Koordination bezieht sich auf eine gezielte Verhaltensänderung aus einer Außenperspektive: Das System wird also von außen, d.h. von außerhalb des Systems, gelenkt. Dabei wird angenommen, dass ein Mächtiger von außen das System koordiniert. Mächtige eines Systems können aber auch innerhalb des Systems dieses steuern. Aus dem Blickwinkel der Akteure basiert der Gegensatz zwischen Koordination als Kontrolle und Kooperation aber auf einer Unterscheidung in aktive, innere und passive, äußere Koordination. Unter Kooperation wird die aktive Beteiligung der Akteure an der Steuerung verstanden, wird aber nicht als Koordination aufgefasst.<sup>37</sup>

Die Definition der Koordination soll sich also auf eine Koordination in einem sozialen System, wie einer Organisationsform, beziehen. Einflüsse von außen oder die Tatsache, ob die Akteure weitestgehend aktiv oder passiv sind, spielen dabei keine Rolle. Es wird eine Perspektive eingenommen, die von den abzustimmenden Subsystemen, Akteuren bzw. Individuen ausgeht.

Durch die Arbeitsteilung des Systems in Form einer Organisationsform müssen Teilsysteme ständig aufeinander abgestimmt werden, da diese Teilsysteme getrennt bestehen und nicht in ein anderes Teilsystem überführt werden können. Dieses „ordnende Zusammen-

---

<sup>36</sup> Der Begriff System ist an dieser Stelle eine allgemeine Beschreibung eines sozialen Gefüges oder Bereichs, das nach außen abgegrenzt ist und aus ähnlichen in einer bestimmten Ordnung/Struktur stehenden Elementen besteht (Hillmann 1994: 857).

<sup>37</sup> Kooperation ist in dieser Perspektive eine Art und Weise der Abstimmung, d.h. Koordination, in einem System.

fassen“ bzw. Abstimmen unterschiedlicher Bereiche ist das Ausrichten auf ein Ziel und keine reine Kommunikation (vgl. Kumbruck 2001: 161).

Man kann also die Koordination als eine gezielte Veränderung der autonomen Dynamik einer Organisationsform verstehen (Mayntz 1987: 94). Folgendes Zitat exemplifiziert dies: „Steuerung zielt, gleichgültig, ob Netzwerke oder andere Systeme gezielt verändert werden sollen, in der Regel darauf, Ereignisse und Interaktionen mehr als nur punktuell zu beeinflussen“ (Sydow, Windeler 2000: 3). Eine „plandeterminierte“ Steuerung von außen ist aber nicht möglich, da „kein soziales System vollständig intentional gestaltet und gesteuert werden“ kann (Müller-Jentsch 2003: 132). Steuern bzw. Koordinieren sind intentionale Handlungen, aus denen auch nicht-intendierte (Handlungs-)Folgen entstehen können (vgl. Giddens 1988: 67ff.).

Vielmehr ist Steuerung oder Koordination, einen sich abzeichnenden Systemzustand an einen gewünschten anzupassen. Die Leitidee der Koordination ist das Anstreben eines idealen Zustands (vgl. Müller-Jentsch 2003: 132), wobei der Ist- mit dem Sollzustand abgeglichen und ggf. eingegriffen wird. Die Koordination ist also ein wechselseitiges aufeinander Abstimmen bzw. Synchronisieren von Interaktionen und Erwartungen, die zu gerichteten Handlungen führt, d.h. sie ist eine Institution mit spezifischen Regeln. Dieses aufeinander Abstimmen von Interaktionen beinhaltet Entscheidungen für ein (Teil-)Ziel oder einen Sollzustand und wird in einem sozialen Aggregat gefällt. Folglich sind Entscheidungen mit der Koordination verbunden (vgl. Messner 1994: 577). Damit werden Entscheidungen durch einzelne Akteure, ein Kollektiv an Akteuren oder kollektivierte Akteure gefällt. Diese Akteure beeinflussen durch die Entscheidung die Handlungen aller oder Teile der Akteure eines sozialen Systems. Kollektiv und kollektiviert heißt in diesem Zusammenhang, dass die Abstimmung kollektiv mit und kollektiviert ohne direkte Akteursinteraktion stattfinden kann, d.h. Handlungen können auch durch gegenseitige Beobachtungen abgestimmt werden. Dabei folgen die Entscheidungen spezifischen, der jeweiligen Koordinationsform eigenen Regeln.

#### 3.2.3.2. Definition der Koordinationsformen anhand von Leitideen

Die Leitidee der Koordination ist ein wechselseitiges Abstimmen, um von einem Ist- zu einem Sollzustand zu gelangen. Die Koordinationsformen beschreiben verschiedene Arten, wie dieses Abstimmen erfolgt. Sie sind verschiedene Arten der Koordinationsinstitution, die sich durch eine unterschiedliche Leitidee und dazugehörige Regeln unterscheiden.

So kann das Aufeinanderabstimmen in den arbeitsteiligen Organisationsformen entweder über ein *Gegeneinander* im Sinne eines *Wettbewerbs* oder eines *gemeinsamen Handelns bzw. Miteinanders* im Sinne einer „*Kooperation*“ bzw. „*Hierarchie*“ erfolgen (vgl. Granados, Gurgsdies 1999: 352ff.). Diese allgemeine Definition des Miteinanders kann wiederum in ein vertikales und ein horizontales Miteinander unterteilt werden. Das vertikale Miteinander bezeichnet die hierarchische, die horizontale eine gleichgestellte, kooperative Steuerung dieser Arbeitsprozesse.

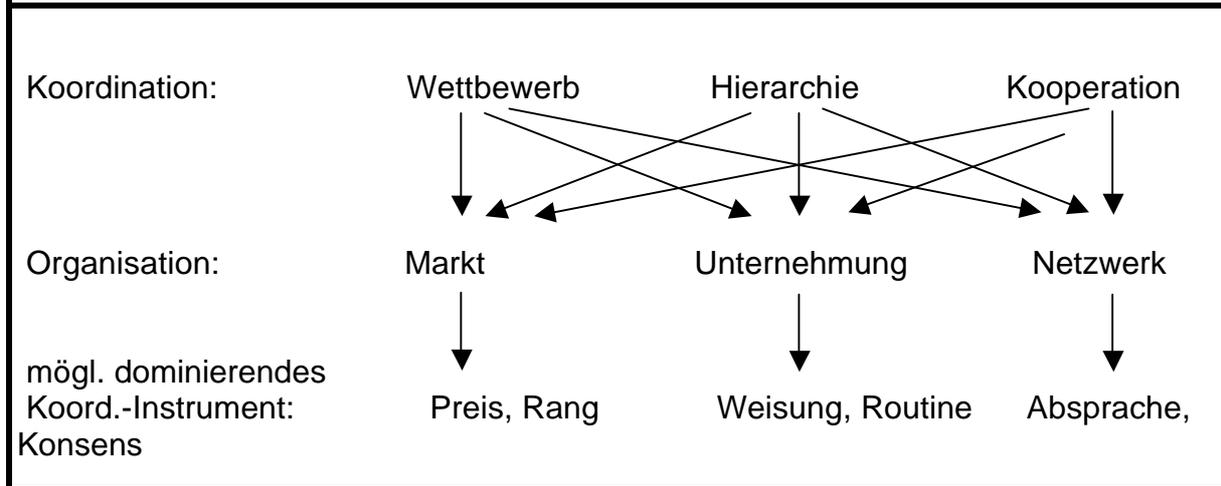
Das Gegeneinander sowie die vertikale und horizontale Kooperation können als drei Koordinationsformen bezeichnet werden, die jeweils eine Leitidee hinsichtlich der Entscheidungen und wechselseitigen Abstimmung beinhalten: Eine Entscheidung erfolgt beim Wettbewerb unabhängig, atomistisch, bei Kooperation gemeinsam, alle umfassend und bei Hierarchie bzw. Macht von wenigen.

Im Weiteren verwende ich, um eine begriffliche Verwirrung zu vermeiden, die verbreiteten Begriffe Hierarchie bzw. Macht für das vertikale und Kooperation für das horizontale Miteinander (vgl. Kavai, Schmid 2004; Becher et al. 2007). Wettbewerb, Hierarchie/Macht

und Kooperation als Koordinationsformen bestimmen die wechselseitige Abstimmung z.B. bei der Zusammenarbeit.

In Anlehnung an die Befunde in Kapitel 3.1. *treten alle drei Koordinationsformen in einer spezifischen Mischung in allen drei Organisationsformen auf* (s. Abb. 3-1). Dabei kann eine spezifische Koordinationsform in einer Organisationsform dominieren (s.a. Kap. 3.2.2).

Abb. 3-1: Organisationsformen der Koordination; Kawai, Schmid 2004: 253; s.a. Becher et al. 2007 (leicht verändert)



Da der Fokus auf der Koordination in der Handlungsebene liegt, interessiert aber die jeweilige Ausprägung der vorhandenen Instrumente einer Koordinationsform und die jeweilige Stärke bzw. Wichtigkeit des Instruments. Für eine Analyse der Softwareentwicklung in dem Arbeitsnetz Open Source-Projekt wird daher die Ausprägungen bzw. Elemente der Koordinationsformen im Folgenden anhand eines Kategorisierungsschemas definiert.

Bei den Koordinationsformen wie Kooperation oder Hierarchie gibt es zwar eine dezidierte Auseinandersetzung mit Elementen der Koordinationsformen im Diskurs über Organisationsformen, aber keine allgemeine, begründete Definition des jeweiligen Begriffs unter Berücksichtigung aller Elemente und ihres gegenseitigen Bezugs (s. Kap. 3.1.). Bei Wettbewerb wird dagegen immer direkt von Marktkonkurrenz gesprochen, aber eine dezidierte Auseinandersetzung mit dem Begriff und den Elementen des Wettbewerbs erfolgte nur rudimentär (vgl. Hillmann 1994; Fuchs-Heinritz et al. 1995; Arentzen, Lörcher 1997: 4360, 4369ff.). Aus diesem Grund strebe ich auf der einen Seite eine *weitestgehend allgemeine, abstrakte, übergreifende Definition der Koordinationsformen* an, die auf alle Organisationsformen angewendet werden kann. Die Koordinationsformen Kooperation, Hierarchie und Wettbewerb treten zwar in allen Organisationsformen auf, unterscheiden sich aber jeweils (s.u.). Der Unterschied besteht in der jeweiligen Ausprägung der Koordinationsform, aber auch im Stellenwert der Koordinationsform. In einer Organisationsform kann daher eine spezifische Ausprägung einer Koordinationsform dominieren. Auf der anderen Seite wird wegen der Spezifika der Organisationsformen die Definition der Koordinationsformen *auf das Arbeitsnetz Open Source-Projekt bezogen*, d.h. die Definition soll möglichst die empirisch auftretenden Handlungen als Normen bzw. Elemente der Koordinationsformen identifizieren helfen. Wird z.B. ein Markt untersucht, treten in dieser Organisationsform andere Instrumente, Regeln bzw. Mechanismen der Koordinationsformen auf, die entsprechend angepasst werden müssen (s. Becher et al. 2007 für den Arbeitsmarkt). Bei der Definition wird außerdem auf die Koordinationsformen als statische Institutionen geachtet. Der Pro-

zess der Institutionalisierung, Genese bzw. des wechselwirkenden Umgangs mit den Koordinationsformen wird anders als z.B. bei der Betrachtung der Macht bei Paris (1998) ausgeklammert.

#### 3.2.3.3. *Kategorisierungsschema zur Einordnung von Koordinationsformenelementen*

Die Koordinationsformen wurden anhand ihrer Leitideen definiert. Weitere Strukturelemente werden aber in der Literatur hinsichtlich Kooperation, Hierarchie/Macht und Wettbewerb diskutiert, wobei ein zentrales Koordinationsinstrument in Anlehnung an die Leitidee einer Koordinationsform eine Rolle spielt. Damit aus den verschiedenen Koordinationselementen eine passende Definition entsteht, werden u.a. die Diskurse über Markt, bürokratische Organisation (Unternehmen) und Netzwerk nach möglichen Elementen durchsucht. Ein weiteres Augenmerk liegt auf verschiedenen Diskussionszusammenhängen der Ökonomie, Soziologie, Politologie, Sozialpsychologie und Ethnologie, in denen Kooperation, Hierarchie/Macht und Wettbewerb als soziale Grundkategorien thematisiert werden. Diese Diskurse beziehen sich kaum aufeinander, es werden aber ähnliche Vorstellungen der Koordinationsformen verwendet. Wegen der Masse an Forschung zu Kooperation, Hierarchie/Macht und Wettbewerb wird bei der Definition dieser Koordinationsformen eher eklektisch vorgegangen und auf ähnliche disziplinäre Konzepte gesetzt, die dann in ein Kategorisierungsschema einsortiert werden. Dabei orientiert sich die Auswahl der zitierten Autoren eher an einer späteren empirischen Anwendung bei einem Softwareentwicklungsprozess mit Zusammenarbeit in dem untersuchten Open Source-Projekt als Arbeitsnetz. Die Definitionen des Marktwettbewerbs passt z.B. kaum zur später untersuchten Organisationsform Arbeitnetz bzw. Open Source-Projekt. Leitend ist aber auch eine Fülle an disziplinären Erkenntnissen, die einer allgemeinen Definition entgegenkommen, unabhängig von Elementen einer Organisationsformendefinition. Für eine eindeutige Einordnung der Elemente der Koordinationsformen wird im Folgenden ein Kategorisierungsschema konstruiert.

#### Herleitung eines Kategorisierungsschemas

In diesem Unterkapitel entsteht ein Kategorisierungsschema, das die Elemente der Koordinationsformen aus den verschiedenen Diskursen ordnet. Bisher wurden die Kategorisierungsschemata und ihre Dimensionen der Organisations- bzw. Koordinationsformenmerkmale eher eklektisch und heuristisch festgelegt (s. Kap. 3.1; vgl. Wiesenthal 2005: 232).

Ein neu zu entwickelndes Kategorisierungsschema soll dabei helfen, die Definition der Koordinationsformen möglichst überschneidungsfrei zu halten. Damit soll ermöglicht werden, die Koordinationsformen bei der empirischen Untersuchung von sozialen Situationen, hier Teilprozessen der Softwareentwicklung, unterscheiden zu können. Zusätzlich soll das Kategorisierungsschema gewährleisten, dass die Definitionen der Koordinationsformen, die in den obigen Diskursen verschiedene Betrachtungsperspektiven und unterschiedliche Erkenntnisgegenstände haben, begrifflich auf einer gleichen Ebene liegen. Das Kategorisierungsschema sollte aber auch einer inneren Logik folgen, die die einzelnen Kriterien aufeinander bezieht. Außerdem beinhalten die Koordinationsformen als Institutionen eine Leitidee, die sich in einem Kriterium widerspiegeln soll.

Da die Koordination mit ihren Entscheidungen mit Handlungen zusammenhängt, wurde besonders im handlungstheoretischen Bereich gesucht und ein Kategorisierungsschema gefunden: All diesen vorgenannten Kriterien entspricht eine übliche handlungstheoretische Unterscheidung mit den Dimensionen „Handlungsbedingungen, Handlung und Handlungsergebnis“ (vgl. Homann, Suchanek 2005: 40, 91ff.; Giddens 1988; Messner 1994:

577). Unterstützend wirkt, dass Entscheidungen in die Koordinationsformen inkludiert sind und dass dieses Kategorisierungsschema auch bei der Entscheidungs-/Spieltheorie genutzt wird (vgl. Laux, Liermann 2003; Holler, Illing 1996).<sup>38</sup>

In einer vorläufigen Definition sind Handlungsbedingungen Regeln, die vom Akteur nicht direkt beeinflussbar und eine notwendige Voraussetzung für das Vorliegen einer Koordinationsform sind. Handeln als Dimension beinhaltet die Leitidee (der jeweiligen Koordinationsform) als Mechanismus, d.h. die Regel, die im Handeln erkennbar ist und mit der eine Koordination durchgeführt wird. Das Handlungsergebnis stellt die Folgen der Handlung einer Koordinationsform für die interagierenden Akteure dar. Mit diesem Kategorisierungsschema können die drei Koordinationsformen Kooperation, Hierarchie und Wettbewerb konzeptualisiert werden.

Üblicherweise wird vor allem in der Ökonomie das Kategorisierungsschema Handlungsbedingungen, Handlung und Handlungsergebnis folgendermaßen gesehen: In der Transaktionskostentheorie (vgl. Williamson 1985, 1993; Göbel 2002) und der Spieltheorie (vgl. Homann, Suchanek 2005; Holler, Illing 1996) z.B. wirkt eine statische Struktur äußerer Institutionen als Transaktionskostenatmosphäre oder Handlungsbedingung auf eine Transaktion als Handlung. In der Industrieökonomik wird die Struktur eines Marktes durch seine Umwelt bestimmt, wobei diese das Verhalten der Unternehmen beeinflusst, woraus sich dann später ein Resultat ergibt (s. Krol, Schmid 2002: 83f.). Bei der Entscheidungstheorie wird von Umweltzuständen als statischer Struktur ausgegangen, die nicht nur die Eigenschaften des Entscheiders, sondern auch seine organisationsinterne und –externe Umwelt enthält. Der Entscheider folgt dabei einer Entscheidungsregel, die aus der Zuordnung von Präferenzen zu Entscheidungsalternativen anhand eines Optimierungskriteriums besteht und zu einem Ergebnis führt (Laux, Liermann 2003: 38ff.).

Diese Sichtweise ist aber begrenzt, da damit eine direkte, z.T. zeitlich abfolgende, deterministische Beziehung zwischen Bedingungen und Handlung postuliert wird. Die Trennung in statische Struktur als Bedingung und Handeln als dynamisches Element führt dabei zu Problemen bei der empirischen Betrachtung. Bei der empirischen Untersuchung des elektronischen Arbeitsmarkts bestand z.B. eine solche angenommene Beziehung bei Koordinationsformen nicht und erschwerte somit die Interpretation der empirischen Ergebnisse (vgl. Becher et al. 2007).

#### Definition des Kategorisierungsschemas

Giddens Theorie der Strukturierung versucht dagegen, Handeln und Struktur zusammen zu betrachten, was einer empirischen Untersuchung über reales Verhalten, bei der beide Aspekte auftreten, eher entgegenkommt. Das Handeln wird als ständig ablaufende Praxis gesehen, in der die Regeln wie Koordinationsmechanismen beim Handeln immer wieder reproduziert werden. Damit treten sowohl statische als auch variable, dynamische Elemente im realen Verhalten auf. Ein sozialer Wandel der Strukturen kann z.B. durch nicht-intendierte Handlungsfolgen, durch unbewusste und bewusste Motivation der miteinander interagierenden Akteure oder unerkannte Handlungsbedingungen mit folgenden Handlungsvariationen erfolgen. Bedingungen sind nicht die statische Struktur und die Handlung das Dynamische, sondern die Bedingungen sind Teile der Regeln, die zusammen mit der Regel auftreten (Giddens 1988: 67ff.; 78). Daraus folgt, dass sich Handlungen in sozialen Situationen wie den Teilprozessen der Softwareentwicklung ändern können sowie und das Statische nicht von vorneherein erkennbar ist, wenn die Handlungen beobachtet werden.

---

<sup>38</sup> Ein Beispiel für die Anwendung des Kategorisierungsschemas besteht in der Industrieökonomik (Krol, Schmid 2002: 83f.).

Statische und variable, dynamische Elemente treten im realen Verhalten gleichzeitig auf, der Blick ist in dieser Arbeit aber auf das Statische gerichtet. Aus dieser Perspektive stellen Bedingungen nicht-zeitliche, den Koordinationsmechanismus als Handlung begleitende Basiselemente dar. Bedingungen sind damit nicht andere, vorausgehende Koordinationsformen, die in eine andere soziale Situation gehören und die aufgefundene Koordinationsform festgelegt haben. Ein Beispiel für eine vorausgehende Verabredung ist z.B. die hierarchische Festlegung eines Übungskampfs/-wettbewerbs in einer Übungsstunde eines Sportvereins (vgl. Sader 1991: 137).<sup>39</sup> Dies begünstigt die Untersuchung von Koordinationsformen in sozialen Situationen wie Teilprozessen der Softwareentwicklung, da zeitliche Abfolgen nicht berücksichtigt werden müssen.

Bedingungen müssen aber nicht notwendigerweise vorliegen, um eine Handlung im realen Verhalten zeigen zu können. Nur eine Handlung, d.h. der Koordinationsmechanismus wird reproduziert, die Bedingungen aber nicht unbedingt. Die Beziehung zwischen Handlungsbedingung und Handlung als Koordinationsmechanismus ist vielmehr durch einen gemeinsamen sinnhaften Bezugspunkt wie die Leitidee bestimmt und nicht zeitlicher oder räumlicher Natur. Die Handlungsbedingungen beziehen sich dabei weitestgehend auf die Leitidee des Koordinationsmechanismus. Der Koordinationsmechanismus ist eine Regel der wechselseitigen Abstimmung und Entscheidung, die die Leitidee der Koordinationsform repräsentiert. Der Koordinationsmechanismus als Regel wird dabei als Handlung einer Koordinationsform reproduziert und kann in weitere Unterregeln bzw. Instrumente (z.B. Machtmittel) aufgeteilt werden. Diese Unterregeln bzw. Koordinationsinstrumente können als Ausprägungen der Koordinationsformen aufgefasst werden. In Weiterführung dieser Überlegung können effektive und effiziente Koordinationsformen mit dem Vorhandensein und der Bedeutung der Handlungsbedingungen der Koordinationsformen zusammenhängen.<sup>40</sup> Beim Handeln ist die Koordination als Handlung/Mechanismus eine Regel, die angewendet wird. Bei der Koordination folgen also Interaktionen einer bestimmten Regel bzw. dem Koordinationsmechanismus, wobei die Interaktionen synchronisiert bzw. aufeinander abgestimmt werden.

Das Handlungsergebnis wird als Motivation von den Akteuren angestrebt, wobei mindestens ein Akteur eine Koordination anstreben will, um ein für ihn positives Ergebnis zu erreichen. Es kann intendierte und nicht-intendierte Folgen beinhalten, die aber nur durch Kenntnis der Motivation aller Akteure festgestellt werden können. Das Ergebnis wird aus Vollständigkeitsgründen dargestellt, obwohl die Motivation der Akteure in der Empirie schwierig bis gar nicht festzustellen ist. Außerdem sind die konkreten Ergebnisse von vornherein nicht feststellbar, weswegen ihre allgemeine Erscheinungsform als Verteilung des angestrebten Nutzens (z.B. als Win-Win- oder Win-Lose-Situationen) vorgestellt wird. Die Ergebnisse und Gerechtigkeitsüberlegungen sind dabei unabhängig voneinander zu sehen.

Nach diesem Kategorisierungsschema mit den Dimensionen „Handlungsbedingungen“, „Handlung/Koordinationsmechanismus“ und „Ergebnis“ werden im Folgenden die Koordinationsformen Kooperation, Hierarchie und Wettbewerb untersucht und die Merkmale und Koordinationsinstrumente aus der jeweiligen Literatur einsortiert (vgl. Brand, Schmid 2006). Dabei wird z.T. relativ ausführlich auf einzelne Merkmale eingegangen, da diese Grundlage für eine Verwendung in der Neuen Institutionenökonomie in Kap. 6.1. sind. Für eine allgemeine Definition der Koordinationsformen greife ich sowohl auf Lexikonartikel,

---

<sup>39</sup> Eine zeitlich vorausgehende Handlung kann die nachfolgenden Bedingungen oder Regeln festgelegt und ermöglicht haben. Diese Frage ist aber nicht Teil der Fragestellung und wird deswegen nicht untersucht.

<sup>40</sup> Dieser Gedanke ist Grundlage der Erklärung in Kap. 6.1.

wie auch auf sozialwissenschaftliche, sozialpsychologische und ökonomische Literatur (s.o.) z.B. den theoretischen Netzwerkdiskurs, die Neue Institutionenökonomie, Machttheorien, Reziprozitätstheorien etc. zurück und entnehme diesen einzelne Elemente der drei Koordinationsformen. Da es sich bei dem untersuchten Open Source-Projekt um ein virtuelles Phänomene handelt, bei dem eine virtuelle Kommunikation dominiert, liegt das Augenmerk auf der Koordination über Kommunikation.

#### 3.2.3.4. *Koordinationsform Kooperation*

Bisher wurde Kooperation vor allem im Zusammenhang mit dem Netzwerkdiskurs (s. Kap. 3.1.), aber auch in anderen, z.B. sozialpsychologischen, soziologisch-ethnologischen, spieltheoretischen<sup>41</sup> Diskurszusammenhängen betrachtet (z.B. Sader 1991: 66ff., 136f.; Mauss 1984; Sofsky 1983; Zucker 1986; Holler, Illing 1996). Vertrauen als Mechanismus der Handlungskoordination steht oftmals im Vordergrund (vgl. Bradach, Eccles 1991; Fuhse 2002). Damit wurde oft auch Reziprozität, Interdependenz oder Verlässlichkeit in Verbindung gebracht (vgl. Powell 1996; Mahnkopf 1994; Windeler 2001). In bürokratischen Organisationen hat Max Weber Kollegialität als Kooperation erkannt, aber keinen hohen Status zuerkannt. Dabei spielen auch Gleichheit und Konsens in bürokratischen Organisationen eine Rolle (vgl. Waters 1989).

In der Sozialpsychologie wird unter prosozialem Verhalten, das unschwer als Kooperation verstanden werden kann, das Helfen, Schenken oder Teilen diskutiert und mit Altruismus in Verbindung gebracht. In einer Beziehung mit Kooperation werden auch moralische Verpflichtung, Empathie, Vertrauen, Reziprozität und Selbstwertsteigerung genannt (vgl. Bierhoff, Klein 1992). In dem Rahmen werden auch extrinsische, d.h. durch externe Belohnungen hervorgerufene, und intrinsische, d.h. allein durch Spaß am Verhalten hervorgerufene, Motivationen diskutiert (vgl. Bierhoff 1993). Gerade Empathie als Einfühlung bzw. „sich in andere hineinversetzen können“ ist mit Emotionen verbunden und führt zu einer empathischen Kooperation (vgl. Spieß 1998).

Reziprozität im Sinne einer Gabe mit Gegengabe und Altruismus werden in der Ethnologie und der Soziologie u.a. im Zusammenhang mit Tauschprozessen und Interdependenz diskutiert (Adloff, Sigmund 2005: 214ff.; Sahlins 2005). Altruismus steht dabei mit einem Selbstopfer in Verbindung, das eingegangen wird, um einem Dritten zu helfen, aber nicht eingefordert werden kann. Der Altruismus stellt dabei eine Sonderform der Gabe dar (Adloff, Sigmund 2005: 222), wobei die Gabe durch sozialkohäsive Reziprozität und Normenbezug gekennzeichnet ist (Mahnkopf 1994: 72). Auch aus der Perspektive der experimentellen Spieltheorie wurde erkannt, dass Reziprozität einen Normenerhalt und eine Kooperation ohne finanzielle Belohnungen erklären kann (vgl. Fehr, Gächter 1998).<sup>42</sup> Reziprozität tritt nach Sahlins 2005 aus dem Blickwinkel eines primitiven Tauschs in drei Formen auf, die sich nach ihrer sozialen Nähe, ausgehend von einer Person unterscheiden. Die generalisierte Reziprozität erfolgt innerhalb einer Familie, eines Haushalts, Dorfes mit starkem sozialem Zusammenhalt, wobei von direkten Gegengaben abgesehen wird. Die ausgeglichene Reziprozität besteht eher zwischen Freunden oder Personen des gleichen Stammes, wobei gleiches getauscht wird. Die negative Reziprozität aber bezeichnet eine

---

<sup>41</sup> Die Spieltheorie geht dabei von einer individuellen Entscheidung zwischen Konflikt als Wettbewerb gegenüber Kooperation aus (vgl. Holler, Illing 1996; Homann, Suchanek 2005).

<sup>42</sup> „Das Wesen dieses Arbeitsprinzips heißt: do ut des, ich gebe, damit du mir gibst. Auch die Begriffe des ›Kuhhandels‹, Tauschgeschäfts, Kompromisses und der gegenseitigen Anpassung berühren dieses Arbeitsprinzip ... Das Wesentliche an einem System der zeitverschobenen gegenseitigen Kompensation ist, daß alle Gruppenmitglieder gewinnen können.“ (Sartori 1992: 229–230)

ungestrafte und umsonst erfolgte Inbesitznahme auch durch List und Betrug, die stark an die Kosten-Nutzen-Kalkulation der Neuen Institutionenökonomie erinnert (s. Kap. 6.1.).

Reziprozität, Empathie, Interdependenz, Altruismus, Konsens etc. umschreiben eher Handlungsmerkmale, denn reale Handlungen.<sup>43</sup> Die Verbindung von kooperativem Koordinationsmechanismus als kooperative Handlung mit der Verhandlung, inklusive der Kommunikation und Entscheidung von Zielen, Bedürfnissen und Möglichkeiten, ist zielführender, wobei Kompromisse eine Rolle spielen können (vgl. Semlinger 2003: 79; Niechoj 2003: 252ff.; Messner 1994; Mayntz 1992; Sofsky 1983: 252ff.; Kumbruck 2001).

Im Folgenden werden die genannten kooperativen Elemente in die Dimensionen „Handlungsbedingungen, Handlung/Koordinationsmechanismus und Ergebnis“ einsortiert und in Beziehung zueinander gesetzt.

- Kooperation basiert auf *Handlungsbedingungen* wie gleichgestellten, autonomen Akteuren, die ein gemeinsames Ziel oder ein gemeinsames Interesse haben, das sie u.U. auch aus verschiedenen Motivationen verfolgen. Das gemeinsame Interesse kann dabei von gegensätzlichem Interesse begleitet werden, das in der Spieltheorie als Mixed-Motives-Games bezeichnet wird (Holler, Illing 1996; Homann, Suchanek 2005).

Kooperation ist freiwillig und gleichberechtigt, wobei eine konsensuelle Selbstbindung an die Regeln der Kooperation von allen Beteiligten durch das Anstreben einer Win-Win-Situation und/oder einer gegenseitigen Abhängigkeit vorgenommen wird. Ein Vertrauensvorschuss, der beim Urvertrauen gelernt und bei der erfolgreichen spieltheoretischen Tit-for-Tat-Strategie am Anfang gegeben wird, wird bei einer Kooperation i.A. vorausgesetzt. Auch wechselseitiges Vertrauen bestimmt eine Kooperation (vgl. Sofsky 1983: 254f., 279; Kumbruck 2001: 161; Zucker 1986; Brand 2004; Axelrod 1984). Das Vertrauen kann nach Zucker (1986) und Lane/Bachmann (1996) in institutionenbasiertes, eigenschaftsorientiertes und prozessorientiertes Vertrauen unterteilt werden. Das institutionenbasierte Vertrauen bezieht sich auf eine Organisations-/Systemzugehörigkeit sowie -kontrolle und das eigenschaftsorientierte Vertrauen auf eine gemeinsame Zugehörigkeit zur gleichen sozialen Gruppe z.B. Ethnie mit gleichen Normen/Werten. Beide Vertrauensarten sind ähnlich dem Urvertrauen als Grundvertrauen zu sehen, bei dem anhand von Signalen ein Vertrauensvorschuss gegeben wird, solange Erwartungen an Institution/Organisation und Gruppeneigenschaften erfüllt werden. Dabei bestehen implizit gemeinsame Werthaltungen, auf die eine kooperative Beziehung aufgebaut werden kann (vgl. Reichwald et al. 2004: 58). Prozessbasiertes bzw. kalkulatives Vertrauen bezieht sich auf Erfahrungen und Beobachtungen des Verhaltens über die Zeit z.B. bei wiederholten Beziehungen und erhöht ein Grundvertrauen, wobei die Wahrscheinlichkeit weiteren Verhaltens, der Vertrauenswürdigkeit und des persönlichen Nutzens kalkuliert wird (Noteboom 2002: 12f., 42). Vertrauen kann aber nicht eingefordert werden, da es eine persönliche Leistung ist. Das Vertrauen tritt nicht nur am Anfang einer neuen Kooperation auf, sondern wirkt auch während der Durchführung einer Kooperation als Handlung. Die oben genannten Emotionen, die z.B. beim prozessbasierten bzw. kalkulativen Vertrauen keine Rolle spielen, und Reziprozität in ihren verschiedenen Formen sind weitere kooperative Handlungsbedingungen, die eine Differenzierung des kooperativen Koordinationsmechanismus bestimmen.

- Der *Kooperationsmechanismus* bzw. die *Handlung* der Koordinationsform Kooperation ist das (wechselseitige) Verhandeln bzw. Aushandeln eines gemeinsamen Ziels mit einem Konsens als Verhandlungsergebnis. Fokus ist dabei ein gemeinsames Handeln. Der Mechanismus als Regel bezieht sich darauf, dass alle Beteiligten einmütig zustimmen, also

---

<sup>43</sup> Empathie und Altruismus sind eher Motivationen für Kooperation, Konsens, Reziprozität und Interdependenz meinen eher Bezüge zwischen Handlungen.

Einstimmigkeit als Entscheidung vorliegt. Die Einstimmigkeit hängt dabei mit Interdependenz zusammen. Die reine Kommunikation kann aber nicht als Kooperation aufgefasst werden, sondern ist nur ein wichtiger Teil. Damit ist ein „zwangloser Zwang eines besseren Arguments“ verbunden (Habermas 1971: 137; vgl. Kumbruck 2001: 161; Sofsky 1983: 253f.; Schimank 2000).<sup>44</sup>

Eine Kooperation muss aber nicht immer inhaltlich ausgehandelt werden, sondern kann auch übersituativ, in der untersuchten Situation auf vorherigen Abmachungen als formale oder informale Institutionen zurückgreifen (vgl. Sofsky 1983: 255, 279). Dabei steht Kooperation mit den z.B. extern gegebenen sozialen Normen in Verbindung. Konsens als Interessenkongruenz kann dabei ohne aufwendiges Verhandeln durch sofortiges wechselseitiges Vertrauen bzw. Vertrauensvorschuss entstehen und umgekehrt (vgl. Göbel 2002: 122; Möllering 2001). Oder kurz gesagt: Ein Verhandeln ist dann nicht nötig, wenn ein Konsens durch Vertrauen vorweggenommen wird. Das Vertrauen ersetzt die Aushandlung dabei graduell. Der Vertrauensvorschuss mit der Annahme eines gemeinsamen Interesses bzw. Ziels wird gegeben, wenn die Akteure entweder einer gleichen sozialen Gruppe angehören (eigenschaftsbasiertes Vertrauen), mindestens ein Akteur einem bürokratisch-organisationalen Verband (z.B. Unternehmen, staatliche Verwaltung) angehört oder bestimmte institutionalisierte Signale Anwendung finden (System-/institutionenorientiertes Vertrauen)<sup>45</sup> oder die Akteure sich aus vorherigen Interaktionen kennen (kalkulatives/prozessbasiertes Vertrauen). Letzteres, kalkulatives/prozessbasiertes Vertrauen tritt wiederum bei der Aushandlung eines gemeinsamen Ziels bzw. eines Konsenses durch wiederholte Kontakte bzw. Reziprozität auf (s.a. Zucker 1986; Lane, Bachmann 1996; Williamson 1993).

Ein Konsens wird durch Reziprozität bzw. wechselseitiges Geben und Nehmen bei der diskursiven Konsensfindung erreicht. Die Koordinationsinstrumente und der Ressourceneinsatz sind mit Reziprozität verbunden, wobei weiteres (prozessbasiertes) Vertrauen erworben werden kann (vgl. Zucker 1986). Eine Person kann zwar dabei eine andere schädigen, doch wird eine Kooperation normalerweise beendet, wenn diese Schädigung entdeckt wird. Davon ausgehend können inspiriert von Sahlins (2005), Noteboom (2002: 63-65) und Blau (1964) drei Kooperationsinstrumente der Reziprozität unterschieden werden. Die drei idealtypischen Kooperationsinstrumente unterscheiden sich nach ihrer Entfernung im sozialen Raum, wobei die spezifischen Determinanten, wie Äquivalenz, Zeit, Raum und Person, sich mit steigender altruistischer bzw. intrinsischer Motivation immer weiter auflösen. Unter intrinsischer Motivation wird die direkte Bedürfnisbefriedigung über die Tätigkeit bzw. Beziehung selbst verstanden. Die intrinsische Motivation wird in spaßbasierte und pflichtenbasierte Motivation unterteilt (vgl. Osterloh 2002). Pflichtenbasierte intrinsische Motivation und Altruismus hängen zusammen, da beide stark durch Normen beeinflusst werden.

Bei der *strategischen Kooperation* als erstes Kooperationsinstrument steht die reine Zielerreichung im Vordergrund, und durch die konsensuelle Verhandlung des Preises oder von anderen, wenigen Vertragsinhalten wird Äquivalenz erreicht. Das gemeinsame Interesse

---

<sup>44</sup> Auch der Rational Choice-Vertreter Schimank (2000: 173ff.; 323ff.) sieht bestimmte Akteurskonstellationen als stabile Strukturen der wechselseitigen Beobachtung, Beeinflussung und Verhandlung auf einer Mikroebene, die unschwer mit Wettbewerb, Hierarchie/Macht und Kooperation gleichgesetzt werden können.

<sup>45</sup> Z.B. ordnet das Signal der Ampel mittels institutionenbasiertem Vertrauen den Straßenverkehr. Bei institutionenbasiertem und eigenschaftsbasiertem Vertrauen sorgen dabei die geltenden formalen und informellen Normen und Sanktionen der sozialen Gruppe oder des Systems/der Vereinigung dafür, dass das Vertrauen erhalten bleibt.

kann dabei für eine bestimmte Situation, kurze Dauer oder einen Teilbereich genau spezifiziert werden (vgl. Neuberger 1998). Weiterhin können flankierende formale oder informale Regelungen eines Gesellschaftssystems oder einer bürokratischen Organisation ein hohes institutionenbasiertes Vertrauen ermöglichen und damit eine strategische Kooperation stark unterstützen: Der Kauf oder Austausch von Gütern findet einmalig, zu einem Zeitpunkt und an einem Ort direkt zwischen zwei Akteuren z.B. auf zwei Marktseiten statt, wobei das beste Beispiel dafür ein Einkauf im Supermarkt ist (vgl. Neuberger 1998; Mahnkopf 1994: 71f.; Pies 2001).<sup>46</sup> Besteht kein flankierendes institutionenbasiertes Vertrauen, muss kalkulatives Vertrauen aufgebaut werden, bevor eine strategische Kooperation erfolgt.

Kommt die strategische Kooperation ohne Empathie aus, ist sie bei reziproker und gruppenbezogener Kooperation wichtig. Die *reziproke Kooperation* als zweites Kooperationsinstrument hängt mit dem erwarteten Austausch von ungefähr gleichwertigen Gütern zusammen. Eine Gegenleistung erfolgt nicht direkt und wird nur bedingt oder in Zukunft erwartet. Dabei fallen Leistung und Gegenleistung räumlich und zeitlich auseinander bzw. Zeit und Raum lösen sich auf, wobei kalkulatives Vertrauen in die Gegengabe eine Rolle spielt. Die direkte Reziprozität entkoppelter und annähernd äquivalenter Gaben dient vor allem der Erhaltung der sozialen Beziehung in Verbindung mit Empathie und von konsensuell ausgehandelten Normen zwischen identifizierbaren Personen oder Gruppen. Die Normen werden zudem oft in situationsgebundenen Aushandlungen bestätigt oder fortgeschrieben (vgl. Stegbauer 2002, 46, 54ff.; Mahnkopf 1994: 72; Mauss 1984; Bourdieu 1998; Blau 1964).

Die *gruppenbezogene Kooperation* als drittes Kooperationsinstrument bezieht sich auf reziproke Beziehungen in einerseits Klein- bzw. Primärgruppen (z.B. Familie), andererseits Groß- bzw. Sekundärgruppen (z.B. Organisationen, wie Sportvereine). Diese selbstorganisierten oder überindividuellen Gruppen bestehen schon und beruhen u.U. auf gesellschaftlichen, ethnischen, nationalen Merkmalen oder selbstproduzierten Logos/Namen etc. Grundlage ist eine generalisierte Reziprozität innerhalb der Gruppe, wobei von der Person, der Zeit (z.B. Generationen) und vom Raum abstrahiert wird. Eine Person bekommt eine Gabe, da sie im Prinzip gleichberechtigtes Mitglied dieser Gruppe ist und sich alle mit der Gruppe auch emotional identifizieren bzw. den Gruppenmitgliedern vertrauen. Dabei spielen institutionenbasiertes und eigenschaftsbasiertes Vertrauen mit durch gruppenbezogene Kontrolle und Signalisierung des Vertretens gleicher Normen/Werte eine Rolle. Die Gegenleistung ist oft nicht äquivalent, da eine direkte Zuordnung der Gaben nicht möglich ist (vgl. Stegbauer 2002: 79). Es bestehen eher „implizite“ Verhandlungen in Form von Empathie, die in Zusammenhang mit Kommunikation von z.B. Hilfsbedürftigkeit stehen, d.h. ein Gruppenmitglied fordert Hilfe an oder die Hilfsbedürftigkeit eines anderen Mitglieds wird erkannt. Die Kommunikation führt mittels intrinsischer Motivation zu einem gemeinsamen Interesse, d.h. zur Hilfe, wobei der Gebende einen Konsens mit dem Hilfesuchenden annimmt. Dabei wirken Kooperationsnormen wie Solidarität innerhalb der Gruppe (vgl. Stegbauer 2002: 79, 93ff.).

Als besondere Form der gruppenbezogenen Kooperation ist die altruistische Kooperation zu nennen, die besonders bei sehr großen Gruppen mit großer sozialer Distanz auftritt. Die gemeinsame Gruppenzugehörigkeit basiert daher auf wenigen gemeinsamen Attribu-

---

<sup>46</sup> Strategische Kooperation kann auch längere Zeit zwischen Über- und Untergeordneten (Vorgesetzten und Mitarbeitern im Unternehmen) auftreten, muss aber immer wieder aktiviert und kontrolliert werden, da die gemeinsamen Interessen auf eher kurze, einmalige Situationen bezogen sind. Etwas anders ist es bei Lehrer-Schüler-Beziehungen, bei denen reziproke Kooperation vorherrscht, da auch eine emotionale Bindung besteht und Normen die Beziehung bestimmen.

ten der Gruppe oder einer imaginierten Zugehörigkeit. Eine imaginierte Großgruppe ist die Nation, eine „letzte“ die Menschheit. Es wird keine Gegenleistung erwartet, weswegen eine äquivalente, zeitlich und räumliche Gegengabe keine Rolle spielt. Neben der Kooperations-/Solidaritätsnorm wirken Emotionen wie Mitleid bzw. Empathie bei der Gabe von Leistungen, was sich in anonymen Kleinspenden widerspiegelt (vgl. Adloff, Sigmund 2005: 221f.; Stegbauer 2002, 98ff.).

Das *Ergebnis* einer Kooperation ist eine Win-Win-Situation, d.h. die beteiligten Akteure erzielen zusammen einen Gewinn, den sie als Einzelne zusammengerechnet nicht erreicht hätten. Entweder werden alle Akteure besser gestellt; es kann sich aber auch mindestens eine Person besser stellen, z.B. bei Altruismus, wobei die Koordinationsregel gleich bleibt (Meeker 1971: 490, 493). Das Erreichen einer Win-Win-Situation kann mit einem Kompromiss verbunden sein, wenn entgegengesetzte und gemeinsame Interessen auftreten. Dabei führt ein Kompromiss zu einem Konsens<sup>47</sup>: Bei einem Kompromiss passen die Akteure ihre (maximalen) Ziele bzw. entgegengesetzten Interessen an die anderen Beteiligten an und nehmen einen Aufwand in Kauf bzw. stecken zurück, um eine Win-Win-Situation auch geringeren gemeinsamen Nutzens zu erreichen.

### 3.2.3.5. *Koordinationsform Hierarchie/Macht*

Hierarchie/Macht wurde im Rahmen des Diskurses über Organisations- bzw. Koordinationsformen mit Unternehmen, staatlichen Organisationen etc. (bürokratischen Organisationen) gleichgesetzt (s. Kap. 3.1.), aber auch in anderen, z.B. sozialpsychologischen, soziologischen Diskurszusammenhängen betrachtet (z.B. Sader 1991: 66ff., 136f.; Grunwald, Lilje 1981; French, Raven 1959; Weber 1980; Hradil 1980).

Macht bzw. Hierarchie sind diffuse und allgemein gehaltene Begriffe (Buschmeier 1994: 8; vgl. Gukenbiehl, Kopp 2003). Im Allgemeinen wird Hierarchie/Macht im Zusammenhang mit Autorität, sozialem Einfluss, Gewalt und Herrschaft diskutiert. Autorität meint die Überlegenheit, die einer Person von anderen bzgl. Merkmalen oder Ressourcen zugeschrieben wird, wobei diese anderen nach Anerkennung von dieser überlegenen Person streben. Der Begriff sozialer Einfluss bezieht sich allgemein auf eine Wirkung, die eine Person über Macht auf andere hat, wobei kein Zwang, aber eine „weiche“ Einwirkung auf die beeinflusste Person im Einklang mit ihrer Einstellung und ihren Werten erfolgt. Gewalt gegen Personen oder Sachen bezieht sich auf ein Machtmittel mit spezifischer Wirkung, wenn Gesundheit, Persönlichkeit, Leben oder Besitz eines anderen beeinträchtigt oder zerstört werden. Herrschaft hängt mit Legitimität zusammen, wobei der Herrschende um die Glaubwürdigkeit seines Machtanspruchs wirbt und versucht, dadurch Machtstrukturen zu verstetigen (vgl. Fuchs-Heinritz et al. 1995: 410f.; Gukenbiehl, Kopp 2003; Popitz 1986; Buschmeier 1994: 14ff.; Weber 1980)

Hierarchie/Macht wird im Zusammenhang mit bürokratischen Organisationen (s. Buschmeier 1994: 51ff.) entweder als strukturbezogene Komponente mit formalen Normen bzw. statischen Routinen und sehr starker Regelgeleitetheit auf der einen Seite (vgl. Weber

---

<sup>47</sup> Die Unterscheidung zwischen direktem Konsens und Kompromissuche ist auch in der Spieltheorie zu finden. In kooperativen Spielen haben alle Beteiligten ein gemeinsames Interesse, das verfolgt wird. In nicht-kooperativen Spielen bestehen entgegengesetzte Interessen, die über Verhandlungen zu einem Kompromiss („informalen oder formalen Vertrag“) führen, wobei am Schluss ein Konsens besteht (vgl. Holler, Illing 1996).

1980; Weise 2000) oder als dynamische, prozessuale, informale, unregelmäßige Beeinflussung auf der anderen gesehen (vgl. Popitz 1986, Friedberg 1995).<sup>48</sup>

Die formalisierte und statische Variante (Hierarchie) steht mit einer Reduktion der Ausführungsmöglichkeiten durch Standardisierung, Bürokratisierung und Algorithmisierung nach rationalen Kriterien in bürokratischen Organisationen wie Unternehmen in Verbindung (Dose 2005: 22f.). Hierarchie wird dabei als eine Pyramidenstruktur von Über- und Unterordnungsverhältnissen in Form von legitimierten Posten ausgehend von der Spitze bis zur Basis in einer bürokratischen Organisation identifiziert (vgl. z.B. Hillmann 1994: 332). Die formalen Normen oder statischen Routinen stellen eher einen Ausdruck der legitimierten Hierarchie und der bürokratischen Organisation dar, da diese durch die Rangordnung, von oben, über Anweisungen als Machtmechanismus vorgegeben wurden. Herrschaft ist in diesem Fall die legitimierte, verstetigte Postenstruktur, wobei die Unterlegenen am Bestehen der Hierarchie/Macht durch den Glauben an die Rechtmäßigkeit der Macht mitarbeiten (Weber 1980).

Macht in ihrer dynamischen, prozessualen, informalen, unregelmäßigen Variante bezieht sich aus einer Tauschperspektive auf Unstetigkeitsstellen/Handlungsspielräume in bürokratischen Organisationen, wobei aus der Perspektive der formalen Netzwerkanalyse Ressourcenbesitz und Informationsmacht/Beeinflussungsmacht eine Rolle spielt (Friedberg 1995: 109ff.; Jansen 1999: 161). Soziologische und sozialpsychologische Theorien untersuchen aus diesem Blickwinkel verschiedene Elemente der Macht wie Machtmittel bzw. –prozesse, potenzielle und reale Macht und Ressourcen (Buschmeier 1994: 4f.; vgl. Hradil 1980).

Die dynamische, prozessuale, informale, unregelmäßige Macht steht im Fokus der Definition von Hierarchie/Macht als Koordinationsform. Auf die Begriffe Bürokratie, Routinisierung und Formalisierung wird weitestgehend bei der Definition von Hierarchie und Macht verzichtet, da sie zur Definition der bürokratischen Organisation als Organisationsform gehören, aber in dieser Arbeit die Softwareentwicklung in Arbeitsnetzen als Netzwerke untersucht werden.

Macht wird als Handlung untersucht, was die dynamische, informale Variante bevorzugt. Ganz auf die formalisierte und statische Variante der Macht kann aber nicht verzichtet werden, da aus der empirischen Forschung (s. Kap. 3.1.4.1.) legitimierte Posten z.B. als fokale Unternehmen oder Akteure in Unternehmensnetzwerken, Innovationsnetzwerken oder virtuellen Unternehmen bekannt sind und daher auch im Open Source-Projekt auftauchen können. Prozesse der Herrschaftsbildung und der Legitimierung von Herrschaft werden aber ausklammert, wie sie Popitz (1986) beschreibt und Max Weber (1980) mit seinen Herrschaftsformen analysiert. Legitimierte Posten als Elemente der formalen und routinisierten Hierarchie können also nicht vollständig ausgespart werden, sind aber kein Mechanismus der Machtausübung. Legitimität kann folglich als Ressource betrachtet werden, die Legitimierung eines Machteinsatzes oder der Errichtung bzw. Anerkennung von Posten als Herrschaftsformen bleibt aber ausgeblendet. Aus diesem Grund werden beide Begriffe Macht und Hierarchie in einem Atemzug als Hierarchie/Macht und synonym betrachtet. Die ähnlichen Begriffe Autorität, sozialer Einfluss, Gewalt und legitimierter Posten als Herrschaftsformen (s.o.) treten dabei bei der Definition von Hierarchie/Macht als Koordinationsform in den Dimensionen Handlungsbedingungen, Handlung und Ergebnis auf.

---

<sup>48</sup> In der sozialpsychologischen Forschung wird z.B. beim autoritären Führungsstil als hierarchisches Element in einer Organisation Ziel und Weg der Aufgabenerfüllung ständig vorgegeben und kontrolliert (Rosenstiel et al. 1995: 274f.).

- *Handlungsbedingungen* sind bei Hierarchie/Macht der Zugriff einer oder mehrerer Personen auf Ressourcen, wobei sich Hierarchie/Macht nach den unterschiedlichen Zugriffsmöglichkeiten auf Ressourcen unterscheidet. Durch den aktuellen oder potenziellen Einsatz von Ressourcen können Entscheidungen und damit Handlungen beeinflusst werden (vgl. Hradil 1980; Coleman 1991). Dabei verfolgt eine Person oder Personengruppe als Mächtige ein Ziel, das gegenüber den Beeinflussten durchgesetzt wird. Eine Abhängigkeit zwischen Mächtigen bzw. Ressourcenbesitzern und Machtlosen, Beeinflussten bzw. Ressourcenarmen besteht, da verschiedene Personen unterschiedlichen Zugriff auf Ressourcen haben, aber auch andere diese Ressourcen benötigen oder meinen, diese zu benötigen. Folglich sind die Machtlosen vom Ressourceneinsatz anderer abhängig, weswegen diese Machtlosen ihre Ziele, Verhaltensweisen und u.a. Werte an das Geforderte anpassen (Neuberger 1998: 41).

Zwei Arten von Ressourcen werden eingesetzt: einerseits persönliche Ressourcen und andererseits bereits legitimierte Posten als Ressourcen, die durch ein Kollektiv geschaffen wurden (vgl. Hradil 1980: 62ff.; Buschmeier 1994: 51ff.). Persönliche Ressourcen können z.B. ökonomisches, symbolisches, soziales oder kulturelles Kapital, aber auch individuelles körperliches und geistiges Vermögen sein (vgl. Bourdieu 1987, 1998: 108). Posten dagegen sind „objektiv‘ zurechenbare soziale Merkmale“ und unabhängig von der Person (vgl. Hradil 1980: 63). Bei Posten ist der Ressourcenzugriff auf persönliche, aber auch auf überindividuelle, z.B. allgemein geteilte oder besondere Ressourcen durch den zugewiesenen Posten bei anderen legitimiert. Diese besonderen Ressourcen einer virtuellen Organisationsform sind z.B. die IuK-technische Infrastruktur. In einem sozialen Aggregat z.B. der untersuchten Organisationsform Netzwerk besteht sowohl in einer freiwilligen Gruppe, als auch einer formellen Organisation eine formelle oder informelle Legitimierung der Posten. Folglich kann ein Individuum auf persönliche Ressourcen und ggf. Ressourcen einer Organisationsform zugreifen, d.h. eine „positionale“ Macht in einer informellen Gruppe bzw. formalen Organisation tritt zur „persönlichen“ Macht (vgl. Wilke, Knippenberg 1992: 351ff.; Sader 1991; Rosenstiel et al. 1995: 118ff.).<sup>49</sup>

Damit ist ein Abhängigkeitsverhältnis in Verbindung mit einem Ressourcenzugriff zugleich ein Über-/Unterordnungsverhältnis. Aus dem Verhältnis können die Handlung bzw. der Mechanismus der Hierarchie/Macht (Machtmittel) entstehen und durchgeführt werden.

- Der Mechanismus bzw. die *Handlung* bei Hierarchie/Macht ist das Gewähren und Entziehen von Ressourcen, die andere benötigen. Hierarchie/Macht ist aber nach Weber (1980: 28) die Chance, innerhalb einer sozialen Beziehung den eigenen Willen oder ein Ziel *auch* gegen das Widerstreben einer oder mehrerer Personen durchzusetzen. Damit besteht eine innerpsychische Komponente im Sinne eines Widerstrebens und eine Verhaltenskomponente im Sinne eines Widersetzens. Zusätzlich können die Interessen des Mächtigen oder der Mächtigen und der Beeinflussten entweder überwiegend übereinstimmen oder überwiegend auseinanderfallen bzw. entgegengesetzt sein (Buschmeier 1994: 13f.; vgl. Coleman 1991; Becher et al. 2007). Eine Entscheidung fällt der Mächtige oder einige Mächtige in seinem/ihrer eigenen Interesse.<sup>50</sup> Der Mächtige gibt eher seinen Willen bekannt, wenn ein Ziel gefördert und durchgesetzt werden soll (Durchsetzungsmacht).

---

<sup>49</sup> Der Ressourceneinsatz sollte nachträglich erklärt werden, wenn weiterhin eine soziale Beziehung z.B. in Abhängigkeit oder in Kooperation bestehen soll. Die Erklärungen sind mit der Stärke der Abhängigkeit des Ressourcenabhängigen verbunden. Wenn keine Erklärung erfolgt, wird der Ressourceneinsatz als willkürlich und aus einem ohnmächtigen Zustand empfunden, was zu Fatalismus, angstbasiertem Entscheidungsstillstand, Gegenmacht mit Revolte oder Flucht/Abbruch der Beziehung führen kann.

<sup>50</sup> Im Weiteren wird von einem Mächtigen gesprochen, obwohl dabei auch mehrere Mächtige vorhanden sein können.

Soll aber ein Ziel verhindert werden (Verhinderungsmacht), äußert sich der Mächtige u.U. nicht. Nicht nur das konkrete Ausführen von Macht als aktueller/manifester Machteinsatz, sondern allein der potenzielle Einsatz eines Machtmechanismus als latente Macht kann schon ein Durchsetzen des eigenen Willens bzw. ein Befolgen nach sich ziehen, da bei Nichtbefolgen eine Sanktion in Form eines Machtinstruments erfolgt (vgl. Hradil 1980: 34ff.). Latente/potenzielle Macht basiert auf einer Willensbekundung, die u.a. mit Drohungen zusammenhängt, d.h. dem in Aussicht gestellten Einsatz von Machtmechanismen. Sind die Sanktionen und der Willen des Mächtigen schon von vornherein bekannt, stellt sich latente Macht als vorausschauendes Gehorsam dar. (An)Weisung oder Befehl als verbale bzw. schriftliche Äußerungen ähneln Drohungen, wobei die Sanktionen nicht explizit sind, sondern implizit mitschwingen. Die möglichen Sanktionen wie Entlassung, Einkommenseinbußen, „Karriereknick“ etc. sind allen Beteiligten bekannt und führen zu Gehorsam (vgl. Paris 1998). Doch treten Weisungen vor allem in bürokratischen Organisationen wie Unternehmen auf, dieser Ansatz lässt sich jedoch nicht ohne weiteres auf Open Source-Projekte als informelle Organisationen übertragen (vgl. Bergquist, Ljungberg 2001: 310).

Bei latenter und manifester Macht ist jeweils eine Kontrolle wichtig (Buschmeier 1994: 25ff.), da festzustellen ist, ob der Machtlose sich nach den Vorgaben bzw. sich im Sinne des Mächtigen verhalten hat. Dabei wird nicht Verglichen und auch kein gradueller, sondern ein dichotomer Unterschied gemacht. Im Allgemeinen wird dabei zwischen einer Verhaltenskontrolle und Endkontrolle unterschieden (vgl. Alchian, Demsetz 1972). Hat sich der Machtlose nicht entsprechend den Vorgaben verhalten, muss dann der Mächtige, um seine Glaubwürdigkeit zu erhalten, sanktionieren bzw. Machtinstrumente einsetzen.

Da Hierarchie/Macht durch Zugriffsgewährung und Entzug von benötigten Ressourcen bestimmt ist, können die Machtinstrumente nach den Ressourcen eingeteilt werden. Abhängig von den Ressourcen können die Machtinstrumente Bestrafen, Belohnen oder Manipulieren (über Identifikation, Expertentum, Information und legitimierte Normen) analytisch unterschieden werden (vgl. Hradil 1980: 42ff.). Zudem werden diese auch von den meisten sozialpsychologischen und soziologischen Autoren genannt (Buschmeier 1994: 24ff.).<sup>51</sup> Werden bei Bestrafen bzw. Belohnen die psychischen, physischen und sozialen Merkmale einer Person und ihre Handlungsmöglichkeiten direkt beeinflusst, erfolgt ein Manipulieren mit indirekter Wirkung auf den Machtlosen überwiegend über eine Einflussnahme auf die Kommunikation und ihre Inhalte. Bestrafen und Belohnen hängen u.a. mit dem Verständnis des Machtlosen zusammen, weswegen ein Machtmechanismus eingesetzt wird. Beim Manipulieren dagegen weiß der Machtlose nicht unbedingt, dass Macht ausgeübt wird.

*Bestrafen* hängt als negative Sanktion mit z.B. physischer und psychischer Gewalt und *Belohnen* als positive Sanktion mit z.B. monetären Anreizen zusammen (vgl. Hradil 1980: 44f.). Lob als Belohnung oder Tadel als Bestrafung eines Beeinflussten durch einen Mächtigen vor Dritten kann auch darunter gefasst werden (vgl. Paris 1998: 152ff.). Die Wegnahme einer Ressource kann aber in einem Fall als Bestrafen, in einem anderen Fall als Belohnen verstanden werden (Buschmeier 1994: 26). Beispielsweise kann ein Straffälliger zu einer Haftstrafe verurteilt werden, wobei ihm die Freiheit als Bestrafung genommen wird, wohingegen durch ein Geständnis die Haftstrafe als Belohnung verkürzt wird. Damit spielt der mit Bestrafen und Belohnen verbundene Sinn von Ressourcengabe/-entzug eine Rolle. Bestrafen kann die Reduktion von Ressourcen sein, auf die ein Mensch angewiesen

---

<sup>51</sup> Die konzeptionellen Machtarten und die eher empirisch gewonnenen Machttaktiken aus der sozialpsychologischen Literatur sind laut Buschmeier (1994) ineinander überführbar bzw. beziehen sich auf das Gleiche.

ist oder die eine Person als gegeben bzw. allgemein üblich ansieht. Belohnen ist dagegen eine zusätzliche Gabe einer Ressource, die über das allgemein Übliche hinausgeht.

Die manipulativen Machtarten bestehen aus den Unterformen Experten-, Informations-, Identifikations- und Legitimationsmacht (vgl. French, Raven, 1959; Raven, 1992; Buschmeier 1994: 25ff.), die u.a. mit den oben genannten Ressourcen/Kapitalarten zusammenhängen. *Expertenmacht* ist einerseits formales Bildungskapital (Zertifikate), das in gesellschaftlichen Bildungsanstalten erworben wird, oder informelles symbolisches Kapital bzw. Reputationsmacht, das in informellen Gruppen u.a. durch Erfahrung gewonnen wird. *Informationsmacht*<sup>52</sup> hängt mit einer „Gatekeeper“-Funktion in einem Kommunikationsnetzwerk zusammen, das durch den Zugriff auf Informationen anderer Personen beruht und daher sozialem Kapital gleicht. *Identifikationsmacht* ist mit dem besonderen Charisma bzw. der Attraktivität einer Persönlichkeit oder u.U. einer Gruppe durch besondere persönliche Ausstrahlung verbunden, wobei sich die anderen, scheinbar Machtlosen, aus einer Ich-Schwäche (vgl. Autorität) heraus, mit dieser Person oder Gruppe identifizieren.<sup>53</sup> Diese ähnelt somit Max Webers charismatischer Persönlichkeit mit ihren Identifikationsangeboten. *Legitimationsmacht* bezieht sich auf die internalisierten Normen des Machtlosen und deren Gültigkeitsbereich. Damit ist keine Amtsautorität als Macht durch legitimierte, allgemein anerkannte Posten gemeint, obwohl die Posten als internalisierte Normen z.B. in einem Unternehmen eine Rolle spielen können. Posten haben in Bezug auf die informellen Arbeitsnetze eher eine höhere Einflussmöglichkeit. Macht wird ausgeübt, indem aufgefordert wird, sich entsprechend den Normen der Gruppe oder Organisation zu verhalten, mit der man sich identifiziert bzw. zu der man sich zugehörig fühlt<sup>54</sup> (Hradil 1980: 45f.).

Die Nutzung der Machtarten ist auch von ihrer Legitimität abhängig, d.h. ob sie zum bestehenden Regelungsgefüge passen oder die Akteure gleiche Interessen haben. Es ist daher zu erwarten, dass bei gleichen Interessen der Mächtigen und der Beeinflussten eher die Machtart Manipulieren und bei unterschiedlichen Interessen eher Belohnen und Bestrafen anzutreffen sind. In den untersuchten Arbeitsnetzen wird wegen einer dominierenden Kooperation mit überwiegender Interessensübereinstimmung wahrscheinlich eher die manipulative Machtart anzutreffen sein.

Das *Ergebnis* der hierarchischen Koordination erhöht auf jeden Fall den Nutzen des Mächtigen, unabhängig davon, ob der Nutzen des Machtlosen auch erhöht wird. Dies hängt von der Interessenslage des Nicht-Mächtigen ab, weswegen zwei Zustände möglich sind: Erstens kann der Nutzen des Machtlosen stagnieren oder erhöht, wenn nicht maximiert, werden, was dem Ergebnis der Kooperation gleichkommt. Zweitens kann der Nutzen des Machtlosen sinken, was der Nutzenverteilung des Wettbewerbs gleicht (vgl. Becher et al. 2007; Meeker 1971: 490). Dabei kann der Mächtige u.a. Anpassung oder Auseinanderklaffen der Interessenslagen beeinflussen wie die Forschung zur autoritären bzw. demokratischen Führung zeigt (Rosenstiel et al. 1995: 274f.). Daher sind alle Nutzenabstufungen zwischen den beiden Zuständen vom Mächtigen ansteuerbar.

---

<sup>52</sup> Experten- und Informationsmacht sind jeweils Teilbereiche der Deutungsmacht.

<sup>53</sup> Für eine genauere Betrachtung von Sachverständigkeit als Expertenmacht oder Schimpfklatsch/Intrige/Seilschaft als Sonderformen der Informationsmacht kann Rainer Paris (1998) herangezogen werden.

<sup>54</sup> Dies steht hinter Ravens (1992) „Legitimate Power“, wobei vor allem hinter der Legitimität von Reciprocity, Equity, Dependence der Rückgriff auf gemeinsame Normen des Mächtigen und des Beeinflussten steht, also der gleichen Gruppe.

### 3.2.3.6. Koordinationsform Wettbewerb

Wettbewerb als solcher wird häufig, ausgehend von Lexikonartikeln und dem Diskurs über Organisations- bzw. Koordinationsformen (s. Kap. 3.1.) mit einer Konkurrenz um Preis oder Quantität/Qualität auf einem Markt gleichgesetzt (vgl. Arentzen, Lörcher 1997: 4360, 4369ff.; Hillmann 1994; Fuchs-Heinritz et al. 1995). Üblicherweise wird zwischen den Akteuren, den Nachfragern und den Anbietern, unterschieden, wobei die jeweiligen Akteure einer Marktseite miteinander im Wettbewerb stehen, wie Arbeitgeber untereinander und Arbeitnehmer untereinander auf dem Arbeitsmarkt (vgl. Pies 2001). Die ökonomische Wettbewerbstheorie analysiert hauptsächlich den Wettbewerb auf dem Markt und vergleicht diesen oft mit einem vollkommenen/perfekten Wettbewerb als effizienteste Form (vgl. Krol, Schmid 2002). Da aber die idealtypische Form des Wettbewerbs und nicht seine effiziente oder faire Form definiert wird, können kaum Annahmen des vollkommen/perfekten Wettbewerbs der mikroökonomischen Theorie, aber allgemeine Annahmen zum Wettbewerb verwendet werden.

Sozialpsychologische, soziologische Diskurse werden zusätzlich herangezogen, um eine idealtypische Wettbewerbsdefinition zu erhalten. In diesen Diskursen wird Wettbewerb in Gruppen oder soziale Beziehungen ohne den ausschließlichen Blick auf perfekten oder fairen Wettbewerb oder auf den Markt analysiert. Damit können auch die wettbewerblichen Handlungen im später untersuchten Open Source-Projekt besser erfasst werden.

In der Sozialpsychologie und teilweise der Soziologie (z.B. Sader 1991: 66ff., 136f.; Grunwald, Lilge 1981; Sofsky 1983), aber auch der Spieltheorie (vgl. Holler, Illing 1996), wird Wettbewerb als Konflikt gesehen. In der sozialpsychologischen Forschung stehen dabei Wettbewerb bzw. Konflikte zwischen und innerhalb von Gruppen im Fokus. In Gruppen bestehen u.a. unterschiedliche individuelle Interessen und eine Konkurrenz um Aufgaben, d.h. es wird um knappe materielle oder symbolische Güter konkurriert (Sofsky 1983: 312f.; 316).

Etwas abseits des herkömmlichen ökonomischen Verständnisses von Wettbewerb als Marktgeschehen mit vielen Beteiligten, ist die Auffassung eines Wettbewerbs als Konflikt in Form eines Kampfs, bei dem auch physische Gewalt ausgeübt wird.<sup>55</sup> Die Kämpfe haben dabei das Ziel, den oder die Gegner zu schädigen, wenn nicht zu vernichten (vgl. Simmel 1903), d.h. es kann ein Wettbewerb um das Überleben bestehen. Dabei ist die Antriebsfeder u.a. eine subjektive Feindseligkeit aus Hass bzw. Wut. Ein Kampf kann auch um ein materielles oder symbolisches Gut z.B. aus Habgier oder zum Überleben erfolgen, das einem der Gegner gehört oder gehören könnte. Die Feindseligkeit geht oft mit einem persönlichen Kampf zweier Kontrahenten einher, der die Person in ihrer Totalität zu verletzen, wenn nicht vernichten sucht (Sofsky 1983: 321ff.). Eine weitere soziale Beziehung mit dem oder den Gegnern ist daher nicht mehr möglich (Simmel 1958: 199; Simmel 1903; Sofsky 1983: 314ff.).<sup>56</sup>

Diese Form des Kampfs als Wettbewerb tritt selten auf, da dieser in der Gesellschaft zivilisiert wurde. Reste des Kampfs als Wettbewerb können im schädigenden Marktwettbewerb gesehen werden (s.u.). Im gesellschaftlich-zivilisierten Wettbewerb besteht im Allgemeinen eine Grenze der Gewalt, d.h. der Ablauf des Wettbewerbs wird durch äußere Regeln festgelegt bzw. „zivilisiert“. Als gesellschaftlich anerkannte Formen des Wettbewerbs sind ein gesellschaftlicher Konkurrenzkampf und ein Sachstreit zu unterscheiden (Simmel

---

<sup>55</sup> Es sei daran erinnert, dass in der Managementliteratur bzw. in –zeitschriften zuweilen der Marktwettbewerb als Kampf bzw. Krieg bezeichnet wird.

<sup>56</sup> Beispiel für einen solchen Kampf ist der Kampf rivalisierender Jugend-/Gangsterbanden. Eine zivilisierte Variante dieses Wettbewerbs ist ein Boxkampf zwischen zwei Kontrahenten.

1958: 200ff.; Sofsky 1983: 316). Die Gegner verhalten sich dabei nicht zu- oder gegeneinander, d.h. keine Kraft wird auf den Gegner verwendet bzw. der Gegner wird als „Gegner“ ignoriert. Die Schwächen und Stärken des Gegners werden zwar erkundet, aber es wird nicht auf diesen direkt eingewirkt (vgl. Simmel 1903).

Beim gesellschaftlichen Konkurrenzkampf erfolgt der „Kampf“ mit einem Wettbewerber um ein Drittes, d.h. nur ein gegnerunabhängiges Gut bzw. Ziel kann von einem Wettbewerber ausschließlich erlangt werden. Die Gegner sind sich dabei oftmals sehr ähnlich in ihren Merkmalen, haben aber antagonistische Interessen (Simmel 1958: 214; Simmel 1903; Sofsky 1983: 316). Beispiele sind auch sportliche Wettkämpfe, andere Kampfspiele und auch die Konkurrenz auf einem Markt.

Bei einem Sachstreit fehlt die persönliche Ebene, aber ein Kampf auf einer sachlichen Ebene mit Sachargumenten und antagonistischen Interessen um ein materielles oder symbolisches Gut oder Ziel erfolgt, das einem der Gegner gehört oder gehören könnte.<sup>57</sup> Streit bedeutet auch, dass gemeinsame (Partial-)Interessen, aber zugleich an bestimmten Punkten antagonistische Interessen bestehen können (Simmel 1958: 201ff., 204f.). Die Abstimmungsproblematik von individuellen, antagonistischen Interessen kann in Gruppen durch eine Wahl nach dem Mehrheitsprinzip gelöst werden. Dabei treten verschiedene Untergruppen mit ihren Meinungen in Wettstreit miteinander (vgl. Sartori 1984, 1992).

Die oben genannten Unterscheidungen und Elemente gehen in die folgende Definition des Wettbewerbs mit Handlungsbedingungen, Handlung und Ergebnis ein, werden aber z.T. in einem anderen Zusammenhang gebracht.

- Eine *Handlungsbedingung* des Wettbewerbs besteht aus mehreren unabhängigen, mindestens zwei Akteuren mit unterschiedlichen knappen Merkmalen bzw. Ressourcen (Sofsky 1983: 316). Voraussetzung ist das Vorhandensein eines knappen Ziels oder Guts, um das nach bestimmten Anforderungen konkurriert wird. Die Anforderungen und Regeln des Wettbewerbs sind vorhanden und den Beteiligten bekannt, da sonst der Wettbewerb einer Lotterie gleicht. Dabei werden diese durch die Handlungen des Gegners oder durch bestimmte externe Normen bestimmt. Die Wettbewerbsbeteiligten können sich daher überhaupt nach den Anforderungen und Regeln ausrichten. Ohne Beteiligung der zu Vergleichenden besteht kein Wettbewerb. Die Wettbewerbsbeteiligten vergleichen sich also mindestens mit den Wettbewerbsanforderungen/-merkmalen. Zusätzlich können sich die Akteure wechselseitig beobachten und reagieren jeweils auf das Verhalten der anderen Akteure (vgl. Schimank 2000: 173ff.; 323ff.). Die eigenen Ressourcen werden optimiert, so dass die Akteure eine starke Ähnlichkeit aufweisen (vgl. Simmel 1958: 214).

Generell ist ein freiwilliger Ein- oder Austritt aus einem Wettbewerb möglich, was sich z.B. in einer aktiven oder passiven Beteiligung darstellen kann. Dabei kann ein Wettbewerber ein angestrebtes, knappes Ziel bzw. Gut oder die Wettbewerbsregeln gegen andere oder gegen eine Inaktivität abwägen. Knapp können materielle Güter wie Geld oder Besitz, aber auch immaterielle, symbolische Güter wie Reputation oder Anerkennung, Zahlungsfähigkeit/Preis, Mengen oder andere Qualitäten z.B. überzeugende Argumente sein. Dabei gilt, dass zwischen knappen Ressourcen und knappen angestrebten Gütern zu unterscheiden ist. Je nach sozialer Situation, kann z.B. Reputation selbst als knappes Gut angestrebt oder aber als Ressource zur Erlangung eines anderen knappen Gutes wie Besitz gebraucht werden (vgl. Luhmann 1987: 521ff.).

---

<sup>57</sup> Bestes Beispiel hierfür ist der Rechtsstreit, der durch Verfahrensregeln festgelegt ist, in denen die antagonistischen Interessen aufeinandertreffen (Sofsky 1983: 319f.). Ein weiteres Beispiel für einen (andauernden) Streit mit konträren Meinungen zwischen zwei Kontrahenten kann ein Konflikt zwischen Auftraggeber und -nehmer sein.

- *Handlung* bzw. Mechanismus des Wettbewerbs ist der Vergleich von Merkmalen, wobei eine Entscheidung bezüglich eines Merkmals eines Wettbewerbers gefällt wird. Ein Wettbewerber wird ausgewählt und dessen Nutzen erhöht sich. Dabei werden die Merkmale der Akteure (Qualitäten oder Quantitäten) im Hinblick auf die Anforderungen verglichen, woraus sich das Ergebnis ableitet. Durch den Vergleich wird dann eine Entscheidung herbeigeführt. Die Beteiligten haben dabei das gleiche Ziel, aber antagonistische Interessen: sie rivalisieren gleichzeitig um die Erfüllung der Anforderungen und um ein Ziel oder knappes Gut. Nur Akteure können miteinander konkurrieren, die angestrebten, knappen Güter können dies nicht.

Ein Kampf ist im Prinzip ein Wettbewerb. Da aber bei einem Kampf, der auf Vernichtung des Gegners abzielt, keine weiteren sozialen Beziehungen wie in der Organisationsform Netzwerk möglich sind, wird dieser Bereich bei der Definition weitestgehend ausgeklammert. Bei einem *direkten Wettbewerb* können die Akteure ihre Gegner bei wechselseitiger Beobachtung willentlich schädigen, um ein Ziel zu erreichen (schädigender Wettbewerb). Schädigen heißt, dass auf die Ressourcen des Gegners Einfluss genommen wird (vgl. Simmel 1903; Sofsky 1983: 314ff.). Ein solcher Wettbewerb besteht z.B. beim Meinungsstreit, der auf einer persönlichen oder sachlichen Ebene abläuft. Zwei oder mehr Parteien wollen ihre Meinung durchsetzen. Die Wettbewerbsregeln sind implizit und basieren auf gängigen Kommunikationsregeln. Dabei ist ein Vergleich über die gegenseitige Kommunikation möglich: Die Akteure versuchen, sich aktiv z.B. durch Prahlern über Großtaten gegenseitig zu überbieten oder zu beeinflussen. Bei dieser Art Konflikt wird versucht, den Ruf des Gegners direkt zu schädigen oder den Streit auszuweiten, so dass der Gegner z.B. aus Zeitnot oder wegen Unglaubwürdigkeit aufgibt (ruinöser Wettbewerb). Alle Wettbewerbsbeteiligten vergleichen sich miteinander und verwenden Energie auf den Gegner.

Bei einem *indirekten Wettbewerb* verwenden die Akteure keine Energie auf den Gegner und seine Schädigung. Die Wettbewerbsregeln wurden informal oder formal von außerhalb eines sozialen Aggregats oder durch die Beteiligten eines sozialen Aggregats als Wettbewerber festgelegt und sanktioniert. Vergleichende und Vergleichene können, müssen also nicht aus gleichen Personen bestehen. Die Genese der Wettbewerbsregeln interessiert aber nicht, sondern ihre Anwendung. Einerseits vergleichen sich Wettbewerber miteinander und optimieren ihre Ressourcen, wenn eine wechselseitige Beobachtung anderer Wettbewerbsbeteiligter möglich ist. Andererseits gibt es einen nicht-gegenseitigen Vergleich, der ausschließlich auf einen Vergleich mit Wettbewerbsanforderungen bzw. einen idealisierten Wettbewerber hinausläuft, wenn keine Informationen über weitere Beteiligte zu bekommen sind. Dabei vergleicht sich z.B. auf dem Arbeitsmarkt der Bewerber mit den Anforderungen einer Stelle.

Die Instrumente des Wettbewerbs sind Regeln des Vergleichs bzw. Vergleichsverfahren, die festlegen, welches Merkmal oder welche Merkmale eines Akteurs am besten auf die Anforderungen passen. Dabei werden die Merkmale der Akteure untereinander und mit den Anforderungen dahingehend verglichen, ob ein jeweiliges Merkmal das schönste, höchste, etc. oder umgekehrt das hässlichste, niedrigste, etc. ist. Damit läuft Wettbewerb auf ein Über- oder Untertreffen von Merkmalen hinaus.<sup>58</sup>

Zwei Verfahren sind zu unterscheiden: Einerseits werden graduelle Unterschiede einer Anforderung bzw. eines Merkmals (z.B. Preis, Zahlungsbereitschaft) bei einer Ausschreibung oder andererseits Unterschiede von Mengen an abzählbaren Einheiten bei einer

---

<sup>58</sup> Der Vergleich beim Wettbewerb unterscheidet sich von der Kontrolle bei Hierarchie/Macht dadurch, dass bei der Kontrolle untersucht wird, ob etwas vorhanden ist, beim Vergleich geht es aber um eine graduelle Einschätzung.

Wahl nach dem Mehrheitsprinzip festgestellt. In beiden Fällen kann auch bei einem Vergleich von mehreren Merkmalen ein gewichtetes oder ungewichtetes Ranking-Verfahren (Präferenzreihenfolge der Alternativen) durchgeführt werden (s. Andel 1983: 49ff.; Laux, Liermann 2003: 31ff., 87ff.). Bei einer Ausschreibung werden in einem Bietprozess mit verschiedenen Versteigerungsarten die graduellen Merkmalsausprägungen verglichen (vgl. Homan, Suchanek 2005: 278ff.). Weitere Vergleiche, die über das Merkmal Menge und das Mehrheitsprinzip<sup>59</sup> funktionieren, erfolgen über Wahlverfahren wie das Condorcet-Verfahren. Die Wahl nach dem Mehrheitsprinzip kann Alternativen (als Sachwahl bzw. eine Form des Meinungsstreits) oder Personen umfassen. Der Vergleich selbst ist bei der Wahl die Abgabe und Zählung der Stimmen sowie die Auswertung. Es findet ein Wettbewerb um z.B. favorisierte Alternativen statt, da verschiedene Personen(gruppen) miteinander konkurrieren und u.U. versuchen, sich gegenseitig zu beeinflussen. Bei einer Personenwahl konkurrieren die Kandidaten miteinander (vgl. Heun 1983; Sartori 1984, 1992).

Das *Ergebnis* in Form von Nutzen wird nur für eine Person oder Personengruppe erhöht, d.h. alle anderen gehen leer aus, oder jeder bekommt seiner Leistung entsprechend eine anteilige Nutzenerhöhung. Als Ergebnis wird von den Akteuren die Maximierung der Nutzendifferenz zwischen den Akteuren angestrebt (vgl. Meeker 1971: 490; Hillmann 1994: 436; Simmel 1903). Es können dabei Negativsummenspiele, wobei der Verlierer mehr verliert als der Gegner gewinnt, aber auch Nullsummenspiele bestehen, wobei der Verlust des Verlierers genauso groß ist wie der Gewinn des Gewinners. Akteure passen in einem Wettbewerb ihre eigenen Merkmale oder Charakteristika an, so dass derjenige den Wettbewerb gewinnt, der den Vergleichskriterien am besten entspricht.<sup>60</sup>

### 3.3. Zusammenfassung des Kapitels

Die Auseinandersetzung mit dem Netzwerkdiskurs, der mit Organisationsformen und einem umfassenderen Diskurs über Koordination im Sinne einer Steuerung zusammenhängt, zeigte: In den Organisationsformen wie Markt, Netzwerk oder bürokratische Organisation (Unternehmen oder Verbände) konnten im empirischen Diskurs neben den jeweilig dominierenden Koordinationsformen Hierarchie, Kooperation und Wettbewerb auch andere Koordinationsformen nachgewiesen werden. Diese Erkenntnis hat bisher im theoretischen Diskurs über Koordination und Organisationsformen kaum Eingang gefunden. Mittlerweile verfolgen Becher et al. (2007) für den elektronischen Arbeitsmarkt und der Sachverständigenrat Gesundheitswesen (2005) für den „Gesundheitsmarkt“ einen ähnlichen Ansatz, wobei der Markt als Organisationsform getrennt von den Koordinationsformen betrachtet wird. Damit kann der üblichen Sichtweise widersprochen werden, dass eine Organisationsform mit einer Koordinationsform gleichzusetzen sei. Vielmehr ist die Organisationsform von den Koordinationsformen zu trennen. Würde z.B. obige Gleichsetzung von Netzwerk mit Kooperation gelten, dürfte im Arbeitsprozess der Softwareentwicklung eines Netzwerks in Form eines Arbeitsnetzes nur Kooperation auftreten. Dies ist aber nicht wahrscheinlich, da auch andere Autoren in Netzwerken Macht und Wettbewerb feststellten.

Da auch in der theoretischen Diskussion bis auf Becher et al. (2007) keine fundamentale, systematische Auseinandersetzung mit Koordination, Koordinationsformen, Organisation

---

<sup>59</sup> Eine hundertprozentige Ablehnung oder Zustimmung kann entweder durch Hierarchie/Macht oder durch Kooperation mit Konsens entstehen, weswegen beide Koordinationsformen dadurch nicht bestimmt werden können.

<sup>60</sup> Wird das soziale Aggregat betrachtet, in dem ein Wettbewerb stattfindet, kann die Ressourcenoptimierung vieler Akteure, obwohl nur ein Akteur gewinnt, zu einer Nutzenerhöhung für das soziale Aggregat führen.

und Organisationsformen erfolgte, wurde dies in dieser Untersuchung angestrebt. Dabei wurde das Begriffsdurcheinander im Zusammenhang mit Netzwerken, aber auch mit anderen Organisationsformen geklärt, indem Organisations- und Koordinationsformen weitestgehend getrennt definiert wurden.

Ein Ausgangspunkt für eine vom theoretischen Diskurs geleitete Neudefinition war die Untersuchung des Arbeitsprozesses der Softwareentwicklung innerhalb der Organisationsform Netzwerk in Form eines Arbeitsnetzes, wobei das Arbeitshandeln auf einer Handlungsebene in den Teilprozessen fokussiert wurde. Institutionen strukturieren das Handeln der Individuen. Koordinations- und Organisationsformen wurden als Institutionen verstanden und sind daher umfassende Bündel von Regeln/Normen, die durch eine Leitidee zusammengehalten werden. In diesem Zusammenhang wurde weiterhin von einem sozialen Beziehungsnetzwerk ausgegangen, das sich in Beziehungsstruktur und in Beziehungsdimensionen innerhalb von verschiedenen Organisationen, aber auch bei verschiedenen Koordinationsformen unterscheidet.

Eine begriffliche, vollständige Trennung zwischen Koordinationsform und Organisationsform gestaltete sich allerdings schwierig, da die gemeinsame Leitidee bei den Organisationsformen, wie beschrieben, die Arbeitsteilung ist. Dabei impliziert die Arbeitsteilung eine Koordination. Die Koordination unterscheidet sich wegen der unterschiedlichen Arbeitsteilung in den verschiedenen Organisationsformen als eigenständige, zielgerichtete, künstliche Institutionen auf der Mesoebene: Idealtypisch besteht im Markt eine atomistische<sup>61</sup>, in einer bürokratischen Organisation (z.B. Unternehmen, staatliche Organisationen oder Non-Profit-Verbände) eine abhängige und im Netzwerk eine selbstorganisierte Arbeitsteilung. Netzwerke bestehen aus autonomen Akteuren, die selbstorganisiert, wiederholt temporär gemeinsam Vorteile über informale Beziehungen realisieren, aber einem begrenzten, exklusiven Zugang unterliegen. Netzwerke umfassen dabei nicht nur bürokratische Organisationen wie Unternehmen, sondern auch individuelle Akteure. Arbeitsnetze gehören zu den Netzwerken, wobei als Vorteil ein gemeinsames Produkt durch gemeinsames Arbeiten angestrebt wird. Folglich kann eine spezifische Ausprägung einer Koordinationsform eine Organisationsform dominieren (vgl. Becher et al. 2007).

Unabhängig von der Dominanz einer Koordinationsform wurden die Regeln des Koordinationshandelns konzeptionalisiert und können in folgenden Kapiteln für punktuelle Teilprozessbetrachtungen des Softwareentwicklungsprozesses verwendet werden. Da die Normen in dem Arbeitsprozess der Softwareentwicklung interessieren sowie Koordination und Koordinationsformen Institutionen sind, können Koordinationsformen in einem Arbeitsprozess z.B. in einem Open Source-Projekt als Arbeitsnetz untersucht werden. Zusätzlich kann festgestellt werden, ob die Koordinationsformen in der konzeptionalisierten Weise auftreten. Damit kann außerdem die Nebenfragestellung zur IuK-technische Infrastruktur weiter präzisiert werden. Es wird gefragt, wie die gekapselte IuK-technische Infrastruktur mit der Koordination der Softwareentwicklung in einem Open Source-Projekt als Arbeitsnetz zusammenhängt, die Koordination unterstützt bzw. wie die Technik genutzt wird.

Koordination als Institution beinhaltet die Leitidee eines wechselseitigen aufeinander Abstimmens bzw. Synchronisierens von Interaktionen und Erwartungen, die zu gerichteten Handlungen führen. Eine Koordination wird in Koordinationsformen durchgeführt, die sich in der Leitidee eines Gegeneinanders oder einer vertikalen bzw. horizontalen Kooperation unterscheiden und Wettbewerb, Hierarchie/Macht und Kooperation genannt werden. Die Koordinationsformen wurden systematisch auf einer Ebene konzeptionalisiert und klar voneinander abgegrenzt, so dass die Steuerungsfrage neu angegangen werden konnte.

---

<sup>61</sup> Atomistisch bedeutet, dass ein Arbeitspaket von einem Individuum erstellt und gehandelt wird.

Die Definition der Koordinationsformen war einerseits vom Arbeitsprozess der Softwareentwicklung in einem Arbeitsnetz wie einem Open Source-Projekt inspiriert, wurde andererseits so weitläufig angelegt, dass auch andere Organisationsformen danach untersucht werden können. Mit diesem neuen Blick werden nicht auftretende Koordinationsformen nicht ignoriert, sondern es kann deren Fehlen bzw. Bedeutungslosigkeit thematisiert und weiter untersucht werden.

Weiterhin wurde bei der Definition der Koordinationsformen ein Kategorisierungsschema verwendet, das einen Bezug zwischen den Elementen der Koordinationsformen herstellt, was z.B. bisher im Organisationsformen- bzw. Netzwerkdiskurs Schwierigkeiten bereitete. Dieses Kategorisierungsschema basiert auf einer häufig verwendeten Aufteilung auf der Handlungsebene, wobei zwischen Handlungsbedingungen, Handlungen und Ergebnis unterschieden werden (s. Tab. 3-2).

Tab. 3-2: Elemente der Koordinationsformen anhand des Kategorisierungsschemas, eigene Darstellung

Kategorisierungsschema	Koordinationsformen		
	Kooperation	Hierarchie/Macht	Wettbewerb
Zentrale Handlungsbedingungen	Vertrauen	Über-/Unterordnungsverhältnis aus unterschiedlichem Ressourcenzugriff	(gegenseitige) Beobachtung mit Ressourcenanpassung
Handlung/Mechanismus	Verhandlung mit Konsens	potenzielle und aktuelle Machtdurchführung mit Kontrolle	Vergleich
Koordinationsinstrumente	strategische, reziproke und gruppenorientierte Kooperation	Bestrafung, Belohnung sowie manipulative Machtarten wie Experten-, Informations-, Identifikations- und Legitimationsmacht	direkter, kampffählicher Wettbewerb (Streit) und indirekter Wettbewerb wie Versteigerung/Ausschreibung und Wahl nach dem Mehrheitsprinzip
Ergebnis	Win-Win-Situation	Win-Win-/Win-Lose-Situation	Win-Lose-Situation

Die Handlungsbedingungen sind keine vorausgehenden oder notwendigen Bedingungen, sondern begleiten die Handlung und beziehen sich auf die Leitidee der Koordinationsform. Die An- oder Abwesenheit der Handlungsbedingungen bestimmen aber u.a. die Effektivität und Effizienz der Handlungen. Zentrale Handlungsbedingungen stellen bei Kooperation Vertrauen, bei Hierarchie Über-/Unterordnungsverhältnis aus unterschiedlichem Ressourcenzugriff und bei Wettbewerb die gegenseitige Beobachtung mit Ressourcenanpassung dar.

Die zentrale Handlung in einer Koordinationsform wird, wie gezeigt, durch eine Regel, den sogenannten Mechanismus, bestimmt, der sich von der Leitidee direkt ableiten lässt. Diese Handlung/dieser Mechanismus steht mit Entscheidungen im Zusammenhang und kann in weitere (Unter-)Regeln, die sogenannten Koordinationsinstrumente, aufgeteilt werden. Ist die Handlung/der Mechanismus bei Kooperation eine Verhandlung mit Konsens, bei Hierarchie/Macht eine potenzielle und aktuelle Machtdurchführung mit Kontrolle, ist diese

bei Wettbewerb ein Vergleich. Koordinationsinstrumente werden als Ausprägungen der Koordinationsformen verstanden. Kooperative Koordinationsinstrumente sind eine strategische, reziproke und gruppenorientierte Kooperation. Bei Hierarchie/Macht sind Bestrafung, Belohnung sowie manipulative Machtarten wie Experten-, Informations-, Identifikations- und Legitimationsmacht Koordinationsinstrumente. Beim Wettbewerb stellen sich die Koordinationsinstrumente als direkter, kampfähnlicher und indirekter Wettbewerb dar, wobei der indirekte Wettbewerb weiter in eine Versteigerung/Ausschreibung und Wahl nach dem Mehrheitsprinzip unterschieden werden kann.

Das Handlungsergebnis bezieht sich auf die Motivation der Akteure, die ohne Gerechtigkeitsüberlegungen auf die angestrebte Nutzenverteilung wie eine Win-Win- oder Win-Lose-Situationen abhebt.<sup>62</sup>

---

<sup>62</sup> Das Ergebnis einer Koordinationsform eignet sich nicht als Merkmal für eine Identifizierung der Koordinationsform bei einer späteren empirischen Untersuchung, wurde aber wegen einer möglichen Verwendung in der Neuen Institutionenökonomie in Kap. 6.1. aufgeführt. Der Grund dafür, dass das Ergebnis bei der Hierarchie nicht von dem Ergebnis, d.h. der Nutzenverteilung, des Wettbewerbs oder der Kooperation (Win-Win-Situation) zu unterscheiden ist, liegt neben der schwierigen Erhebung in einer konzeptionellen Überschneidung. Bei Hierarchie wird der Nutzen für den Mächtigen immer maximiert, wobei der Nutzen des Machtlosen variiert. In einem Fall kann Hierarchie zu einer Win-Win-Situation auch für den Machtlosen führen, in einem anderen zu einer Win-Lose-Situation, in der der Nutzen für den Machtlosen sinkt (s.a. Becher et al. 2007; Serries 2005).

## **4. Konzept und Methodik der empirischen Untersuchung**

Dieses Kapitel beschreibt das konzeptionelle und methodische Vorgehen bei der empirischen Untersuchung in dieser Arbeit. Da die Koordination des Arbeitsprozesses der Softwareentwicklung in virtuellen Arbeitsnetzen bisher nicht analysiert wurde, wird ein virtuelles Arbeitsnetz explorativ untersucht, das ein Softwareprodukt erstellt und daher hoch virtuell ist. Ein Open Source-Projekt wird, wie in Kapitel 2 dargestellt, empirisch betrachtet. Im Folgenden wird nicht nur die Datengenerierung für die Fallstudien (s. Kap. 4.1.), sondern auch ein Bewertungsschema für diese Daten vorgestellt (s. Kap. 4.2.). Das empirische Material wird dann in dem nachfolgenden Kapitel 5 gemäß den Fragestellungen ausgewertet und dargestellt. Dabei wird das Bewertungsschema bei der Betrachtung des Open Source-Projekts KDE angewandt.

### **4.1. Methodisches Vorgehen**

Im Folgenden wird die Auswahl der Untersuchungsobjekte, die Datenerhebung (s. Kap. 4.1.1.) sowie die Datenaufbereitung (s. Kap. 4.1.2.) vorgestellt, die zu dieser Untersuchung herangezogen wurden. Die Daten wurden im Rahmen des von der Deutschen Forschungsgemeinschaft geförderten Forschungsprojekts „Struktur und Funktionsweise elektronischer Arbeitsmärkte“ erhoben. Aus diesem Forschungsprojekt wurden mehrere Artikel bzw. Bücher veröffentlicht, die das gleiche empirische Material weitestgehend unter anderen Fragestellungen verwendeten (s. Becher et al. 2005; Brand, Schmid 2006; Brand, Holtgrewe 2007; Holtgrewe, Brand 2007).

#### **4.1.1. Auswahl des Untersuchungsobjekts**

Eine Auswahl erfolgte nach in Kapitel 2 beschriebenen Vorgaben. Die Koordination der Softwareentwicklung in spezifischen virtuellen Arbeitsnetzen wie dem Open Source-Projekt wird explorativ untersucht, weil bisher eher virtuelle Erwerbsarbeit in Unternehmen im wissenschaftlichen Fokus stand (s. Kap. 2). Es soll anhand empirischer Fallstudien gezeigt werden, dass auch in diesem Untersuchungsobjekt, wie angenommen, Koordinationsformen mit spezifischer Ausprägung und spezifischem Stellenwert nachgewiesen werden können (s. Kap. 3). Zusätzlich wird auf den Zusammenhang zwischen IuK-technischer Infrastruktur und der Koordination sowie zwischen Kunden/Endnutzern und Softwareentwicklern geachtet. Das freiwillige, eher unentgeltliche Open Source-Projekt als hoch virtuelles Arbeitsnetz und sich von Unternehmen unterscheidend wurde ausgewählt (s. Kap. 2).

Diese konzeptionelle Eingrenzung stellt eine Vorselektion für eine in der qualitativen Forschung üblichen Fallgruppenauswahl als Interviewtengruppe dar (Flick 1999: 78ff.). Als konkretes Untersuchungsobjekt wurde aus den Open Source-Projekten das Open Source-Projekt KDE (KDE-Projekt) ausgewählt, aus dem Mitarbeiter interviewt wurden.

Die Wahl fiel auf das KDE-Projekt, da zum einen durch Vorrecherchen die ausgeprägte kooperative Ausrichtung bekannt war, die sich damit besonders stark von der Erwerbsarbeit im Unternehmen unterscheidet. Zusätzlich ist es ein renommiertes, großes Open Source-Projekt, das eine bekannte, recht verbreitete Software erstellt. Zum anderen bestand ein einfacher Zugang zu den Programmierern dieses Projekts, da diese sich im Gegensatz zu anderen Open Source-Projekten stark auf Deutschland konzentrieren. Dadurch war ein umfassender Einblick in den Arbeitsprozess möglich, der der Zentralität des Open Source-Projekts KDE für diese Untersuchung entsprach.

Die Fallstudie als qualitative, nicht-standardisierte, explorative Untersuchungsform wurde verwendet, weil anhand von Beispielen relevante Strukturen und Prozesse detailliert

nachvollzogen werden können. Die Fallstudie stützt sich dabei auf verschiedene Methoden, wobei das leitfadengestützte Experteninterview, wie in den Sozialwissenschaften üblich, im Mittelpunkt stand. Die Akteure des Open Source-Projekts wurden dabei als Experten für ihre soziale Wirklichkeit und für Entscheidungsprozesse des Arbeitsprozesses verstanden (vgl. Liebold, Trinczek 2002: 35ff.). Der halbstandardisierte Interviewleitfaden aus dem Forschungsprojekt „Elektronische Arbeitsmärkte“ (s.o.) konnte wegen der relativen Nähe der Fragestellung zu der Fragestellung in dieser Arbeit verwendet werden.<sup>63</sup> Durch ihre relative Offenheit können die Interviews zudem neue Erkenntnisse generieren, die in den konzeptionellen Vorüberlegungen nicht bedacht wurden (vgl. Liebold, Trinczek 2002: 42). Die Dauer der leitfadengestützten Experteninterviews lag zwischen 60-120 Min., konnte in Einzelfällen aber auch 180 Min. umfassen. Die Interviews wurden auf Tonband aufgenommen und z.T. mitprotokolliert. Der Haupterhebungszeitraum war von Mitte 2002 bis Anfang 2003, weitere Nachinterviews und Emailnachfragen wurden jedoch noch bis Anfang 2006 durchgeführt. Die Interviewpartner wurden nach dem Verfahren des Theoretical Sampling ausgewählt, der schrittweisen Festlegung der Interviewpartner z.B. als Extremfälle nach jeweils neu festzulegenden Kriterien (Flick 1999: 81ff.).

#### **4.1.2. Datenerhebung**

Das genaue Vorgehen bei Experteninterviews und allen weiteren angewandten Methoden wie teilnehmende Beobachtungen, Dokumentenanalyse von Homepages, Geschäftsberichten oder Selbstdarstellungen sowie eine quantitative, standardisierte Befragung wird im Folgenden vorgestellt.

Zur Einarbeitung in die Thematik wurde die Dokumentenanalyse der recht vielfältigen Homepage/Website des Open Source-Projekts KDE angewendet. Zusätzlich wurden 62 als Selbstauskünfte zu wertende Interviews von der Projekt-Homepage inhaltsanalytisch ausgewertet (s. Anhang D). Bei diesen Selbstauskünften, die jede Woche auf der KDE-Website veröffentlicht wurden, wurden ausgewählte Mitglieder aus dem KDE-Projekt von einem Projektmitglied per Email mittels Fragebogen befragt. Später wurden sechs qualitative, leitfadengestützte Experteninterviews geführt (Leitfaden s. Anhang B). Folgende Interviewpartner wurden im Sinne eines Theoretical Sampling ausgewählt (Flick 1999: 81ff.): ein Gründungsmitglied, eine Person aus der Peripherie bzw. ein neuer Teilnehmer, ein stark involvierter Entwickler aus dem inneren Kreis, ein Nicht-Entwickler (Übersetzer und technischer Dokumentar), ein stark involvierter, französischer Entwickler auch aus dem inneren Kreis und eine stark involvierte Entwicklerin, die zwischen Peripherie und innerem Kreis einzuordnen ist. Bis auf den französischen Entwickler waren alle Interviewten Deutsche. Dabei ist zu vermerken, dass drei der Interviewten aus dem Open Source-Promoteam<sup>64</sup> stammten. Zu einem späteren Zeitpunkt wurden noch fünf „Nachinterviews“ bzw. Klärungsgespräche geführt, zweimal mit der Entwicklerin und einmal zusammen mit weiteren Entwicklern. Weiterhin wurden zwei telefonische Interviews mit je einem weiteren Releasekoordinator, d.h. zentralen Entwickler, geführt. Drei Interviews wurden durch weitere Email-Nachfragen begleitet. Weiterhin wurde eine teilnehmende Beobachtung mit informellen Gesprächen auf zwei Linux-Konferenzen und einer eigens vom Projekt organisierten KDE-Entwicklerkonferenz durchgeführt, die nachträglich protokolliert wurden. Bei vom Interviewer angestoßenen Gesprächen wurde auf Einzelfragen aus dem Leitfaden für

---

<sup>63</sup> In dem Forschungsprojekt wurde die Funktionsweise von Arbeitsmarktfunktionen untersucht, die wie aus Kapitel 2 bekannt ist, einen impliziten Arbeitsprozess betrachten. Dabei gingen die Arbeitsmarktfunktionen in die Konzeption der hier untersuchten Teilprozesse der Softwareentwicklung ein. Es wird daher in dieser Arbeit das empirische Material nicht nach der Funktionsweise, sondern auf Koordinationsformen untersucht.

<sup>64</sup> Dieses Team organisiert den Auftritt auf Ausstellungen in Deutschland.

Experteninterviews und weiterführende Fragen eingegangen und hiervon Gedächtnisprotokolle erstellt. Auf einer dieser Konferenzen wurde an die anwesenden Open Source-Entwickler des untersuchten Projekts ein Fragebogen ausgeteilt, dessen Fragen auf ersten qualitativen Interviews basierten. Die quantitativ-standardisierte Befragung wurde zur Klärung weiterer qualitativer, nicht-standardisierter Fragen verwendet. 55 ausgefüllte Fragebögen kamen von den Konferenzteilnehmern zurück (s. Anhang C).<sup>65</sup> Zusätzlich wurde die Kommunikation auf drei ausgewählten Mailinglisten über ein Jahr beobachtet und interessante Emails oder Emailthreads dokumentiert. Außerdem wurden vier nachträgliche Email-Interviews, je drei zur neuen Usability-Mailingliste und eines zum Release, durchgeführt. Um die Interviews im KDE-Projekt in einen Kontext zu stellen, wurden zusätzliche Diskussionen und Gespräche mit einem Mitarbeiter des Gnome-Projekts und des Debian-Projekts geführt.

#### **4.1.3. Datenaufbereitung**

Das empirische Material aus leitfadengestützten Experteninterviews und quantitativer Befragung wurde anonymisiert. Fiktive Namen wurden den aufgezeichneten Interviews zugeordnet. Die Interviews wurden im Anschluss mittels eines inhaltlich reduktiven Transkriptionsverfahren verschriftlicht, da die strukturierenden Handlungsvollzüge, aber nicht eine subjektive Einschätzung interessierte. Die Transkription erfolgte durch den Autor, der wie in diesem Verfahren üblich anhand der Fragestellung und des Leitfadens Aussagen paraphrasierte, aber auch vereinzelte treffende Aussagen wortwörtlich niederschrieb. Die Themen des Leitfadens wurden dabei weitestgehend den Aussagen als Überschriften zugeordnet (s. Liebold, Trinczek 2002: 48ff.). Ähnlich wurde mit Gedächtnisprotokollen aus „Nachinterviews“ bzw. Klärungsgesprächen, Selbstaussagen, Daten von der Homepage/Website, Emails aus den beobachteten Mailinglisten, Emailinterviews und von teilnehmenden Beobachtungen verfahren. Einzelnen Protokollabschnitten wurden die Hauptthemen zugeordnet. Da die Interviews, aber auch teilnehmenden Beobachtungen relativ offen sind, wurden für neue Erkenntnisse neue Überschriften bzw. Oberthemen notiert.

Nach diesem Aufbereitungsschritt wurden die festgestellten Aussagen nach ihrem Themenbezug zuerst den Teilprozessen der Softwareentwicklung (s. Kap. 2.3.) und danach Koordinationsformen (s. Kap. 3.2.3.) zugeordnet.<sup>66</sup> Aussagen wurden auch den Akteuren und der oben hergeleiteten Informations-, Kommunikations- und Produktionsausrichtung der IuK-technischen Infrastruktur zugeordnet (s. Liebold, Trinczek 2002: 48ff.). Danach wurden die Aussagen verdichtet und gingen in die Arbeit ein.

## **4.2. Bewertung der empirischen Daten: Indikatoren für Koordinationsformen**

Die Ausprägungen der Koordinationsformen konnten anhand der Klassifikationen in Kapitel 3 identifiziert werden. Bei der Feststellung, wie bedeutend die Koordinationsformen sind, wurde dagegen anders vorgegangen. Es mussten spezifische Indikatoren entwickelt werden, die den Stellenwert der Koordinationsformen abbilden. Ausgangspunkt für eine Indikatorenentwicklung war, dass Koordinationsformen Institutionen auf der Handlungs-

---

<sup>65</sup> Das sind ca. 5 % der Projektbeteiligten. Dabei gilt, dass es bei KDE zwischen 750 und 800 Entwickler und ein zusätzliches Drittel technische Dokumentare und Übersetzer gibt, wobei der Anteil der technischen Dokumentare und Übersetzer nicht genau feststellbar ist.

<sup>66</sup> Dabei wurden nur Aussagen über die Handlungsbedingung und die Handlung/den Mechanismus mit seinen Instrumenten berücksichtigt, da das Handlungsergebnis aus konzeptionell-theoretischen Gründen nicht verwendet werden konnte (s. Kap. 3.4.).

ebene sind. Institutionen unterscheiden sich in ihrer Institutionalisierung, d.h. Individuen berücksichtigen bzw. befolgen die Institutionen in ihren Handlungen unterschiedlich. Damit sind manche Institutionen wie Koordinationsformen stärker institutionalisiert als andere. Die Institutionalisierungsstärke bzw. der Institutionalisierungsgrad einer Koordinationsform konnte also als Maß für die Bedeutung einer Koordinationsform herangezogen werden. Einen hohen Stellenwert hat daher die Koordinationsform, die das Verhalten von Individuen umfassend bestimmt. Da dies eine qualitative Untersuchung ist, wurde dieser Indikator als ein Vergleichsmaßstab verwendet. Für eine Unterstützung dieses Indikators wurden zusätzliche Argumente aus dem empirischen Material gesucht, die die Bedeutung einer Koordinationsform begründen können.

Um eine Bewertung des empirischen Materials vornehmen zu können, wurde der Indikator Institutionalisierungsgrad im Folgenden anhand von Luhmanns Sinndimension angenähert (s. Kap. 4.2.1.). Dabei wurde keine historisch-begriffliche Auseinandersetzung angestrebt, sondern Dimensionen der Institutionalisierungsstärke kurz vorgestellt, die für eine empirische Anwendung prädestiniert sind. Danach wurden die Dimensionen des Institutionalisierungsgrads für die Untersuchung in dem Open Source-Projekt KDE spezifiziert (s. Kap. 4.2.2.). Schließlich wurde für den Arbeitsprozess in dem Open Source-Projekt KDE festgelegt, wie anhand der Dimensionen des Institutionalisierungsgrads ein Gesamtinstitutionalisierungsgrad bestimmt wird (s. Kap. 4.2.3.). Anhand dieser Indikatoren wird für das Open Source-Projekt KDE in Kapitel 5 die Bedeutung der Koordinationsformen festgestellt.

#### **4.2.1. Institutionalisierungsgrad als Indikator**

Der Indikator Institutionalisierungsgrad besteht aus verschiedenen überschneidungsfreien Teilindikatoren, die am Sinn einer Handlung als Institution ansetzen, da sonst die Institutionen nicht berücksichtigt bzw. befolgt werden können. Nach Luhmann (1987: 111ff.; 566ff.) ist Sinn eine Grundlage für Interaktionen in einer Gesellschaft, wobei sich Sinn jedoch in einer Gesellschaft in drei Dimensionen aufspaltet, die sich als zeitliche, soziale und sachliche Dimension darstellen (s.a. Senge 2005: 120ff.). Luhmanns Definition wird verwendet, da er aus einer Basiskategorie nicht nur die nahe liegenden Raum-Zeit-Dimensionen ableitet, sondern auch eine weitere wichtige Dimension benennt.

Die zeitliche Dimension fragt nach der „Dauerhaftigkeit“ einer erwarteten Regel<sup>67</sup> und die soziale Dimension nach ihrer „Verbindlichkeit“, d.h. nach der Anzahl der Personen, für die die erwartete Regel gültig ist. Die sachliche Dimension bzw. „Maßgeblichkeit“ bezieht sich auf die Relevanz einer Institution für den Einzelnen, d.h. inwiefern sich das reale Verhalten vom erwarteten Verhalten unterscheidet. Dabei geht es darum, wie viel Varianz die Institutionen zulassen. Eine Verhaltensvarianz und damit die Relevanz für den Einzelnen nimmt vom Müssen über das Sollen zum Können/Wollen ab (vgl. Luhmann 1987: 111ff.; 566ff.; Senge 2005: 120ff.; Gukenbiehl 1993). Diese eher allgemeinen Aussagen zum Institutionalisierungsgrad mussten für eine empirische Anwendung auf ein Untersuchungsobjekt operationalisiert werden, so dass die vorgefundenen Koordinationsformen danach bewertet werden konnten.

#### **4.2.2. Institutionalisierungsgrad der Koordinationsformen**

Aussagen zu einem Institutionalisierungsgrad einer Koordinationsform können nicht ohne einen Kontext getroffen werden, da die Handlungen in diesem Kontext erfolgen. Die Di-

---

<sup>67</sup> Regeln werden über eine Zustandsperspektive betrachtet, wobei eine Dynamik auch zur Kenntnis genommen wird. Ausgehend von dieser Zustandsperspektive bietet sich das Verhindern der Regeländerung als ein Maß für den Institutionalisierungsgrad an.

mensionen „Dauerhaftigkeit“, „Verbindlichkeit“ und „Maßgeblichkeit“ benötigen daher einen Bezugsrahmen, der mit dem Untersuchungsobjekt als Handlungskontext zusammenhängt, um Aussagen über die Institutionalisierungsstärke treffen zu können. Den geforderten Bezugsrahmen für den Institutionalisierungsgrad stellt das Open Source-Projekt KDE als untersuchtes Arbeitsnetz dar.

- Der Bezugsrahmen für die *Dauerhaftigkeit* ist die Zeit, seit der das Open Source-Projekt KDE existiert. Ein hoher Institutionalisierungsgrad ist vorhanden, wenn eine Koordinationsform als Institution seit Beginn der Softwareentwicklung (Gründung des Open Source-Projekts KDE) besteht oder lange Zeit vorhanden ist. Geringe Bedeutung hat ein sporadisches Auftreten einer Koordinationsform als Verhalten. Das erste und letzte Auftreten einer Institution ist von Bedeutung, wobei die Institution auch weiterhin noch gelten kann.

- Ähnliches gilt für die *Verbindlichkeit*, bei der der Bezugsrahmen die Mitarbeiterzahl des Open Source-Projekts KDE darstellt. Gilt die erwartete Regel für alle Mitarbeiter, so ist sie stark institutionalisiert und gilt sie für wenige, ist dies umgekehrt.

Beide Indikatoren sind einleuchtend, da die Zeitdimension und die Personenanzahl, für die die Institution gültig ist, keiner weiteren Erklärungen bedarf.<sup>68</sup>

- Bei der *Maßgeblichkeit* ist dies anders, da bei dieser Dimension auf verschiedene Indikatoren wie „Stärke der Internalisierung“, „Grad der Legitimität“, „Härte der Sanktionen“, „übereinstimmende Funktionalität für Verhaltensziele der Handelnden“, „gleiche Ergebnisse der Normeninterpretation der Beteiligten“ und „Stimmigkeit des Normensystems ohne Konfliktentstehung“ zurückgegriffen werden kann (Hillmann 1994: 615). Diese Indikatoren beschreiben die Relevanz einer Norm entweder aus der Perspektive eines Individuums oder als überindividuelle Erwartungen bzw. Aktionen Dritter. Sie sind aber für diese empirische Untersuchung z.T. nicht operationalisierbar.

Aussagen über die „übereinstimmende Funktionalität für Verhaltensziele der Handelnden“ und die „gleichen Ergebnisse der Normeninterpretation“ beziehen sich auf innersubjektive Zustände, die aber zugleich überindividuelle Charakterzüge aufweisen. Über eine Vollerhebung könnten diese Indikatoren festgestellt werden, was jedoch in dieser Studie nicht verfolgt werden konnte. Der Indikator „Härte der Sanktionen“ tritt auch in Form von bestrafenden Sanktionen in der Koordinationsform Hierarchie/Macht auf. Außerdem sagt dieser mögliche Indikator nur etwas über eine nachgelagerte externe Einwirkung auf einen Akteur aus und nichts über dessen Bezug zur Regel. Wegen einer möglichen doppelten Verwendung und einer möglichen Begriffsverwirrung wurde auch dieser Indikator nicht verwendet. Der Indikator „Stärke der Internalisierung“ wiederum bezieht sich auf die Institutionalisierungsstärke als Oberkategorie und ist wegen seiner Abstraktheit nicht operationalisierbar. Der mögliche Indikator „Grad der Legitimität“ ist dagegen eher verwendbar. Er kann über die Aussagen, ob einer Institution gefolgt werden muss, soll oder kann (s.o.; Gukenbiehl 1993), das Verhältnis zwischen erwartetem und realisiertem Verhalten rekonstruieren. Zusätzlich kann der mögliche Indikator anhand der Muss-, Soll- oder Kann-Bewertungen aus dem qualitativen, empirischen Material direkt festgestellt werden. Solche Aussagen waren aber im empirischen Material wegen der Offenheit der Interviews nicht unbedingt anzutreffen.

---

<sup>68</sup> Die Konzeptionalisierung der drei Koordinationsformen beinhaltet z.T. gruppenorientierte und zeitliche Bezüge. Die Koordinationsinstrumente der Kooperation basieren z.B. auf einer gruppenbezogenen und zeitlich/räumlich entzerrten Reziprozität. Reputation als Expertenmacht und die Identifikationsmacht als Instrumente der Hierarchie/Macht haben auch einen Gruppenbezug. Diese Koordinationsinstrumente überschneiden sich kaum mit der Institutionalisierungsstärke, da sich beide auf das untersuchte Open Source-Projekt KDE als Arbeitsnetz beziehen.

Als letzter Indikator wurde daher die „Stimmigkeit des Normensystems ohne Konflikte“ ausgewählt. Dabei gilt, je häufiger ein Verhalten ohne Reaktion Dritter auftritt, desto stärker ist es institutionalisiert. Damit ist eher eine „Frequenz von Verhaltensweisen“ in z.T. wiederkehrenden oder ähnlichen sozialen Situationen im Bezug zur Toleranz der Mitarbeiter für Abweichungen verbunden. Maßgeblich ist daher ein Verhalten, wenn dieses sehr häufig in verschiedenen sozialen Situationen ohne Widerspruch auftritt. Folglich ist die Institutionalisierungsstärke hoch, wenn Verhaltensweisen in verschiedenen Situationen häufig auftreten, ohne dass Konflikte entstehen.

Wie aber mit verschiedenen Verhaltensweisen umzugehen ist, die verschiedenen Koordinationsformen zugeordnet und nach diesen Dimensionen bewertet wurden, wird im Folgenden thematisiert.

#### **4.2.3. Messung des Institutionalisierungsgrads in den Teilprozessen der Softwareentwicklung**

Die Feststellung des Institutionalisierungsgrads anhand des Open Source-Projekts KDE als Bezugsrahmen erfolgte wegen des qualitativen Vorgehens anhand einer Ordinalskala. In diesem Bezugsrahmen wurde empirisches Material anhand der drei obigen Dimensionen mit einer groben Abstufung bewertet, d.h. empirische Aussagen wurden nach hohem, mittlerem bzw. niedrigem Institutionalisierungsgrad als hoch, mittel und kaum bedeutend eingestuft. Die Untersuchung dreier Koordinationsformen legte eine Dreierskala nahe. Da mit qualitativen Daten gearbeitet wurde, wurden Aussagen über die graduellen Unterschiede zwischen den Koordinationsformen vermieden. Es konnten aber die drei Koordinationsformen mindestens nach ihrer Bedeutung in eine Rangfolge gestellt werden. Die Dauerhaftigkeit kann in die Skala „seit Gründung“, „seit längerem“ und „seit kurzem“, die Verbindlichkeit in „alle“, „viele“ und „einige/wenige“ Mitarbeiter und die Maßgeblichkeit in „häufig“, „manchmal/oftmals“, „kaum/selten“ unterteilt werden. Dabei können auch dichotome Aussagen in der Form „war bedeutend“ gegenüber „war nicht/kaum bedeutend“ verwendet werden.

In den Teilprozessen der Softwareentwicklung wurde daher folgendermaßen vorgegangen. Der Gesamtinstitutionalisierungsgrad einer Koordinationsform in einem Teilprozess setzt sich aus den einzelnen Institutionalisierungsstärken der vorgefundenen Koordinationsformenelemente/-regeln zusammen. Dabei steigt die Institutionalisierungsstärke mit der Anzahl der bedeutenden (seit Gründung bestehenden, alle Mitarbeiter umfassenden, häufigen) Koordinationsformenelemente in den Teilprozessen an.

Folglich gibt es Abstufungen des Gesamtinstitutionalisierungsgrads der Koordinationsformen, je nachdem, wie viele hochgradige Dimensionen bei mehreren Koordinationsforminstrumenten vorhanden sind. Somit ist die Koordinationsform hoch bedeutend und dominierend, deren hoch bedeutende Instrumente häufig in einer Situation auftreten. Diese Instrumente sind hoch bedeutend, da diese wiederum durch drei Dimensionen als hochgradig bedeutend bewertet wurden. Von diesen drei Dimensionen konnte jedoch vor allem die Frequenz eines Verhaltens festgestellt werden (s.o.), weswegen die Rangfolge eher nach dieser Dimension erfolgte.

Manchmal war die Anzahl der bedeutenden Koordinationsformenelemente in zwei Koordinationsformen gleich oder ähnlich hoch. In diesen Fällen konnte anhand der Bedeutung weiterer Koordinationsformenelemente die Rangfolge festgestellt werden. Dabei dominierte keine Koordinationsform, sondern die Koordinationsformen waren ähnlich bedeutend.

Hin und wieder ergaben sich „inhaltliche Gründe“ direkt aus dem empirischen Material, beispielweise inwiefern die Bedeutung einer Koordinationsform durch die hohe Virtualität

und die spezifischen Arbeitsbezüge im Open Source-Projekt KDE als virtuelles Arbeitsnetz beeinflusst wird. Ein solcher Grund kann eine geringere Bedeutung einer Koordinationsform erklären. Dabei wird implizit von einer geringgradigen Dauerhaftigkeit, Verbindlichkeit und Maßgeblichkeit ausgegangen. Beispielsweise ist Hierarchie/Macht durch die hohe Virtualität begrenzt, was als eine geringe Dauerhaftigkeit, Verbindlichkeit und Maßgeblichkeit aufgefasst wird. Zusätzliche höher bedeutende Koordinationsinstrumente können die Gesamtbedeutung aber erhöhen.

Die drei Dimensionen „Dauerhaftigkeit, Verbindlichkeit und Maßgeblichkeit“ sollen die vorgefundenen Handlungen einer Koordinationsform messen. Dabei sollen alle drei Dimensionen gleichgestellt in die Bewertung eingehen. Doch konnten jedoch bei dieser qualitativen Herangehensweise Dauerhaftigkeit und Verbindlichkeit in der Empirie kaum festgestellt werden. Auch die Bestandsdauer einer Institution konnte wegen der retrospektiven Befragung nicht immer bestimmt werden. In den Interviews wurden vor allem die besonders wichtigen Institutionen berichtet, die schon lange bestanden. Auch die Anzahl der Personen, für die eine Institutionen gilt, konnte nicht eindeutig festgestellt werden, da die Aussagen meistens auf alle Softwareentwickler bezogen wurden. Eine Differenzierung nach Betroffenen war für die Interviewten kaum möglich. Die meisten verwertbaren Aussagen bzw. Informationen im empirischen Material bezogen sich eher auf die Frequenz des Auftretens einer Institution. Bei der Analyse in Kapitel 5 werden alle drei Dimensionen in Betracht gezogen, wobei aber die Maßgeblichkeit (Frequenz) einen erhöhten Einfluss bei der Datenbewertung hatte.

Nachdem für die Koordinationsformen in den einzelnen Teilprozessen und für den gesamten Arbeitsprozess die Art und Weise der Bewertung festgelegt wurde, werden die Indikatoren in dem folgenden Kapitel angewandt.

## 5. Koordinierung des Arbeitsprozesses im Open Source-Projekt KDE

In den vorherigen Kapiteln wurde die neue Arbeitsform Open Source-Projekt als Arbeitsnetz festgestellt und definiert. Zusätzlich wurde die Organisation von der Koordination getrennt sowie die Koordinationsformen definiert. Nach der Erklärung des methodischen Vorgehens wird an dieser Stelle das Open Source-Projekt KDE empirisch untersucht.<sup>69</sup>

Die Annäherung an das Untersuchungsobjekt Open Source-Projekt KDE erfolgt in mehreren Schritten. Zuerst wird die soziale Open Source-Bewegung als Grundlage und Umgebung des Open Source-Projekts KDE charakterisiert, ohne die das KDE-Projekt kaum zu verstehen ist (s. Kap.°5.1.). Dabei wird das kollektive Produkt und seine Distribution aus der Sicht der sozialen Open Source-Bewegung betrachtet. Weiterhin sind die verschiedenen Akteure in der Open Source-Bewegung im Fokus, wobei Open Source-Projekte, darunter das KDE-Projekt, als Akteure in der sozialen Open Source-Bewegung auftreten.

Dieser eher literaturgestützten Charakterisierung und Einordnung schließt sich die empirische Untersuchung des Open Source-Projekts KDE an. Eine Charakterisierung des Open Source-Projekts KDE erfolgt in den Unterkapiteln 5.2. und 5.3. anhand der Projektmitglieder als individuelle Akteure und der verwendeten IuK-technischen Infrastruktur, die einen Einfluss auf die Koordinierung der Softwareentwicklung im Open Source-Projekt haben können.

Die konkrete Koordination der Softwareentwicklung im Open Source-Projekt KDE durch Koordinationsformen als Institutionen steht im Unterkapitel 5.4. im Mittelpunkt. Zuerst wird der Stand der Forschung zu Koordinationsformen in der Literatur über Open Source-Projekte und danach des Zusammenhangs zwischen IuK-technischer Infrastruktur und Koordinationsformen in der Literatur zu Organisationen aufgearbeitet.

Den wichtigsten Teil des Kapitels stellt die nachfolgende empirische Analyse der Koordinationsformen in der Softwareentwicklung im KDE-Projekt dar. Die vorgefundenen Ausprägungen und Bedeutungen der Koordinationsformen werden in den Teilprozessen vorgestellt. Dabei werden nur die Softwareentwickler betrachtet sowie deren Beziehung zu projektexternen Endnutzern. Zusätzlich kann aus dieser Untersuchung mit Fokus auf die Koordinationsformen die Wechselbeziehung der IuK-technischen Infrastruktur mit den Koordinationsformen im Open Source-Projekt KDE herausgearbeitet werden, d.h. wie die Infrastruktur die Koordinationsformen unterstützt bzw. die Koordinationsformen sich der Infrastruktur bedienen.

### 5.1. Annäherung an Open Source-Projekte über die Open Source-Bewegung

In dieser Untersuchung steht das Open Source-Projekt KDE (K Desktop Environment) im Fokus, das zu dem übergreifenden Phänomen einer sozialen Open Source-Bewegung gehört. Für ein besseres Verständnis des Open Source-Projekts KDE erfolgt in diesem Kapitel eine erste grobe Annäherung und Einführung an die soziale Open Source-Bewegung, indem ihre wesentlichen Merkmale herausgearbeitet werden. Die Open Source-Bewegung ist eine soziale Bewegung und unterscheidet sich kaum von anderen sozialen Bewegungen. Eine wichtige Besonderheit der Open Source-Bewegung ist die Wichtigkeit des erstellten Softwareprodukts, das innerhalb von Open Source-Projekten erstellt wird. Eine weitere Besonderheit ist die rechtlich abgesicherte Distribution von Software

---

<sup>69</sup> Für eine historisch-deskriptive Darstellung der Entwicklung des Projekts s. Brucherseifer 2004.

unter spezifischen Softwarelizenzen, die aus gemeinsamen Grundüberzeugungen entstanden. Nach diesem eher allgemeinen Überblick über die Open Source-Bewegung werden die einzelnen Akteure der Open Source-Bewegung vorgestellt. Open Source-Projekte, darunter das KDE-Projekt, spielen als Akteure eine wichtige Rolle in der Open Source-Bewegung und haben durch ihr Produkt eine spezifische Funktion aus Sicht der Open Source-Bewegung. Weitere Akteure werden vertieft vorgestellt, da diese die Umgebung der Open Source-Projekte darstellen, worauf u.a. in Kapitel 5.2. Bezug genommen wird.

Im folgenden Unterkapitel wird das oben genannte Open Source-projektübergreifende Phänomen einer sozialen Open Source-Bewegung untersucht.

### **5.1.1. Open Source-Bewegung als soziale Bewegung**

Ein wesentliches Merkmal des Phänomens einer sozialen Open Source-Bewegung ist, dass diese einerseits anderen sozialen Bewegungen ähnelt, andererseits spezifische Besonderheiten umfasst. In diesem Unterkapitel werden die Ähnlichkeiten zu anderen sozialen Bewegungen und damit die Zugehörigkeit der Open Source-Bewegung zu den sozialen Bewegungen anhand folgender allgemein akzeptierter Kategorien aufgezeigt. Diese Kategorien sind informale Interaktionsnetzwerke, gemeinsame Überzeugung mit Solidarität, „Collective Action“ im Zusammenhang mit Konflikten auch um kollektive bzw. öffentliche Güter und Nutzung von Protest als Druckmittel (vgl. Della Porta, Diani 1999: 15f.; s.u.).

Die Einheiten bzw. Elemente der Bewegung bestehen aus einer Pluralität von Individuen, informalen Gruppen/Cliquen und formalen, bürokratischen Organisationen, die über informale Interaktionen vor allem über Mailinglisten im Internet miteinander vernetzt sind (Della Porta, Diani 1999: 118). Früher standen die sozialen Bewegungen auch schon über nationale Grenzen hinweg miteinander in Verbindung, doch überwogen die national-institutionellen Bedingungen des Heimatlandes (vgl. Dackweiler, Schäfer 1999; Brand 1998: 35). Heute haben sich die verschiedenen informalen und formalen freiwilligen Organisationen im Internet vernetzt und stimmen sich über transnationale Nicht-staatlichen Organisationen ab (Beisheim, Zürn 1999). Dabei unterscheidet sich die Open Source-Bewegung wenig von der heutigen Anti-Globalisierungsbewegung. Die diffuse Beteiligung gestaltet eine Grenzziehung zwischen internem und externem Bereich der sozialen Bewegung schwierig und wird u.U. durch die Internetnutzung noch verstärkt (vgl. Della Porta, Diani 1999: 14ff., chap. 5; Rucht et al. 1997: 126ff.). Da diese Abgrenzung schwierig ist, wird auf die auftretenden vernetzten Einheiten der sozialen Open Source-Bewegung in Kapitel 5.1.2. eingegangen.

Die Vernetzung zwischen den Einheiten der Open Source-Bewegung wird wie in anderen sozialen Bewegungen durch die gemeinsame Überzeugung und Solidarität aufrechterhalten (vgl. Della Porta, Diani 1999: 14ff.). Das untersuchte Open Source-Projekt KDE ist eines von vielen Projekten neben einzelnen individuellen Akteuren in der sozialen Open Source-Bewegung. Alle Projekte und individuellen Akteure fühlen sich über eine gemeinsame Überzeugung und Solidarität mit der Bewegung zugehörig. Die gemeinsame Überzeugung, die die soziale Bewegung zusammenhält, besteht aus dem Gedanken, dass der Quellcode von Software offen zugänglich sein sollte. Dahinter liegt die gemeinsame Ansicht, dass ein freier Fluss von Informationen oder Wissen bzw. ein Zugriff darauf vorhanden sein sollte (s.u.; vgl. für eine Auseinandersetzung mit den Idealen der Bewegung s. Zimmermann 2004). Um diesen freien Fluss von Informationen oder Wissen zu gewährleisten, spielen die softwareproduzierenden Open Source-Projekte und nicht wie in anderen sozialen Bewegungen die protestorganisierenden Einheiten eine herausragende Rolle (s. Kap. 5.1.1.1.). Dabei kann die Softwareentwicklung von nicht-proprietär-kommerziellen

Alternativlösungen auch als Protest verstanden werden. In anderen sozialen Bewegungen wird vor allem auf eine Bewusstseinsänderung, d.h. Änderung der gesellschaftlich-vorherrschenden Werte, durch Proteste und Kritik an Missständen gesetzt (vgl. Della Porta, Diani 1999: 15, 165ff.).

Die gemeinsame Überzeugung aller Einheiten der sozialen Bewegung speist sich aus der ingenieurwissenschaftlichen Fachkultur der 1960er und 1970er Jahre (vgl. Holtgrewe, Werle 2001). In dieser Zeit erstellten Ingenieure und Wissenschaftler gemeinsam in wissenschaftlichen Zentren Software, die noch nicht als verkäufliches, aber als kollektives Gut gesehen wurde. Diese war sozusagen eine „Dreingabe“ zur allerdings sehr teuren Hardware. Später, in den 1980er Jahren, versuchte Richard Stallman dem allgemeinen Trend entgegenzutreten, dass geschriebener Quellcode proprietär, d.h. über eine Lizenz für die Nutzung von Binärcode, verkauft wird. Aus diesem Grund gründete er die Free Software Foundation (FSF) und entwickelte eine spezifische Lizenz zum Schutz des freien Zugangs zu Software bzw. Wissen als „kollektives bzw. öffentliches Gut“ (für eine ausführliche Beschreibung der historischen Entwicklung s. Grassmuck 2002). Damit kann er als ein Gründervater der sozialen Bewegung gesehen werden, der als einer der ersten auf den gesellschaftlichen Konflikt zwischen Software bzw. Wissen als öffentliches Gut und ihre Proprietarisierung aufmerksam machte. Damit ist das kollektive Vorgehen der sozialen Bewegung gegen den fortschreitenden sozialen Wandel einer proprietär-rechtlichen Vereinnahmung durch Unternehmen gerichtet (vgl. Della Porta, Diani 1999: 15; Rucht et al. 1997). Das Besondere an diesem Fall ist, dass eine soziale Bewegung relativ schnell rechtlich die Software als öffentliches Gut direkt und umfassend schützen konnte (s. Kap. 5.1.1.2.). Andere Bewegungen wie die Ökologie- oder Frauenbewegung verteidigen die kollektiven Güter Umwelt und Würde/Gleichberechtigung der Frauen, haben aber trotz ihres längeren Bestehens einige Zeit benötigt, ihre kollektiven Güter rechtlich abzusichern und das auch nur bedingt erfolgreich (vgl. Della Porta, Diani 1999: 226ff.; Nave-Herz 1993).

Die soziale Open Source-Bewegung nutzt zusätzlich einen spezifischen Protest bei ihrem Vorgehen gegen diese spezifische Verrechtlichung. Der Protest zeigt sich nach Zimmermann (2004) in der Benutzung der Lizenz, in der das proprietäre Eigentumsrecht ausgehebelt wird. Der Protest besteht aber eher aus Diskussionen im Internet, Petitionen, Lobbyarbeit, Boykotten und seltenen illegalen Serverattacken, thematischen Konferenzen und seltenen Aktionen, bei denen massenweise Emails an Regierungen industrialisierter Länder bzw. an die Europäische Union versendet werden. Diese Aktionen richten sich z.B. gegen die Ausweitung der Copyrightdauer oder gegen die Einführung von Softwarepatenten (vgl. Williams 2002; Hippel, Krogh 2003: 3ff.; Holtgrewe, Werle 2001: 52). Damit unterscheidet sich die Open Source-Bewegung kaum von anderen sozialen Bewegungen. Da aber die Open Source-Bewegung hoch virtuell ist, traten bisher keine Demonstrationen auf (vgl. Della Porta, Diani 1999: 15, 165ff.; Rucht et al. 1997). Die Softwareentwicklung von nicht-propietären Alternativprogrammen kann jedoch auch als Protest gesehen werden.

### *5.1.1.1. Open Source-Bewegung und ihr Produkt*

Aus anderen sozialen Bewegungen ist bekannt, dass innerhalb der Bewegung spezifische Dienstleistungen entstanden, die den gesellschaftlichen Konflikt überbrücken bzw. mildern sollten. Die Frauenhäuser der Frauenbewegung, die selbstverwalteten Betriebe der Friedens- und Umweltbewegung oder die Arbeiterbibliotheken der Arbeiterbewegung als Dienstleistungen der Bewegungen spielten aber kaum eine zentrale Rolle. Dagegen waren Diskussionszirkel wie in der Frauenbewegung in Form der Selbsterfahrungsgruppen am wichtigsten (vgl. Beywl 1987; Heider, Mevissen 1991; Nave-Herz 1993: 73, 78ff.; Della Porta, Diani 1999). Die spezifischen Dienstleistungen in sozialen Bewegungen wie in der

Open Source-Bewegung konzentrieren sich auf die Produktion einer Dienstleistung oder auf die Unterstützung des Diskurses auch außerhalb der Bewegung.

Das Besondere an dieser sozialen Open Source-Bewegung ist, dass die Produktion von Software oder anderen Dienstleistungen in den Projekten bzw. Gruppierungen einen sehr hohen Stellenwert hat. Mit dem Softwareprodukt soll die Überlegenheit der freien Informationsweitergabe aufgezeigt werden, wobei über die Art und Weise des Aufzeigens die Meinungen in der Bewegung auseinandergehen (s. Kap. 5.1.2.1.). Zusätzlich wird über die IuK-Technik ein öffentliches Gut für alle Menschen zur Verfügung gestellt. Neben dem wesentlichen Merkmal der Softwareproduktion, das auf virtueller Zusammenarbeit mit einer spezifischen Entscheidungsstruktur in einem Projekt beruht, existieren aber noch virtuelle Diskussionszirkel auf Diskussionsplattformen und andere Akteure (s. Kap. 5.1.2.).

Auch Unternehmen wie die oben genannten selbstverwalteten Betriebe zur Open Source-Bewegung produzieren Software unter einer bewegungskompatiblen Softwarelizenz und unterwerfen sich der Kultur der Open Source-Bewegung. Die Mitglieder der sozialen Bewegung und der Projekte arbeiten dabei voluntaristisch, aus eigenem Antrieb, d.h. mit intrinsischer Motivation (s. Kap. 5.2.2.). Bei den Unternehmen hingegen dominiert Gewinnstreben (vgl. Luthiger 2004, Pal, Madanmohan 2002; Leiteritz 2002).

#### *5.1.1.2. Open Source-Bewegung und rechtliche Sicherung der Produktdistribution*

Eine weitere Besonderheit gegenüber den anderen sozialen Bewegungen ist, dass das Softwareprodukt bei der Distribution rechtlich über eine Softwarelizenz als öffentliches Gut abgesichert wird. Neben dieser spezifischen Softwarelizenz gibt es noch weitere Softwarelizenzen, wobei die Lizenzen die Verfügungsrechte unterschiedlich auslegen.<sup>70</sup> Aus der unterschiedlichen Auslegung der Verfügungsrechte können drei Arten von Softwarelizenzen unterschieden werden: die proprietären Lizenzen, die General Public License und andere „Open Source“-Lizenzen.

Die Software bei proprietären Lizenzen ist vollkommen Eigentum des Eigentümers und darf nicht verändert werden, weswegen diese nur in maschinenlesbarer Form vertrieben wird (closed source). Der Eigentümer gibt z.B. nur das Nutzungsrecht für einzelne oder mehrere Nutzer der Software gegen einen monetären Gegenwert frei. Vollkommen geschlossene Software wird nur vom Eigentümer verwendet, Shareware ist nur temporär und/oder gegen eine Spende frei verfügbar und Freeware darf kostenfrei verwendet werden (Jaeger, Metzger 2002: 5f.).

Demgegenüber legt die General Public License (GPL) mittels einer trickreichen Anwendung der Verfügungsrechte fest, dass der Quellcode beliebig weitergegeben und verändert werden darf, dass aber der veränderte Quellcode bei der Weitergabe wieder unter derselben Lizenz veröffentlicht werden muss. Die Lizenz ist ein Garant, dass eine Open Source-/Freie Softwareerstellung weiterhin erfolgt und wieder den vorhergehenden Softwareentwicklern bzw. dem Open Source-Projekt zugeführt wird. Damit soll eine proprietäre Aneignung des freiwillig geschriebenen Quellcodes mit nachfolgendem Einfordern von Lizenzgebühren ausgeschlossen werden. Die Veränderung ist erwünscht, das Ausgrenzen von Personen unerwünscht. Das Eigentumsrecht wird hierbei zur rechtlichen Absicherung eines öffentlichen Gutes genutzt, wobei die Lizenz den offenen Zugang zum Quellcode regelt (vgl. Jaeger, Metzger 2002; Grassmuck 2002). Eine weitere weniger restriktive Li-

---

<sup>70</sup> Die Verfügungsrechte bestehen aus dem Recht auf Gebrauch einer Sache (Usus), dem Recht, Gewinn aus Vermietung und Verpachtung einzubehalten (Usus fructus), dem Recht zur Veränderung der Sache (Usus abusus), dem Recht zur Übertragung aller oder einzelner Rechte auf andere und dem Recht, andere von der Nutzung auszuschließen (Göbel 2002: 66).

zenz ist die Lesser General Public License (LGPL), die überwiegend für Softwarebibliotheken verwendet wird. Der Quellcode wird ähnlich zur GPL offengehalten, kann aber in proprietärer Software verwendet werden.

Die GPL und die LGPL gehören zu den Open Source-Lizenzen<sup>71</sup>, d.h. sie entsprechen rechtlichen Standards, die konform mit den Überzeugungen der sozialen Open Source-Bewegung sind. Andere Lizenzen, die auch zu den Open Source-Lizenzen zählen, stehen zwischen proprietären Lizenzen und der GPL, wobei sich diese Lizenzen durch abgestufte proprietäre Einschränkungen der Verfügungsrechte unterscheiden. Bei einigen Lizenzen ist die Offenheit nur auf die Gegenwart beschränkt und kann in proprietäre Software mit Lizenzgebührrforderung eingebaut werden (vgl. Holtgrewe 2004a: 14; Jaeger, Metzger 2002; Grassmuck 2002: 275ff.). Mit dieser Offenheit gehen aber auch Gefahren einer Übernahme (hijacking) bzw. Verhinderung der voluntaristischen Softwareentwicklung einher (vgl. Lerner, Tirole 2002).

### **5.1.2. Akteure der Open Source-Bewegung, darunter Open Source-Projekte**

In verschiedenen sozialen Bewegungen bestehen unterschiedliche informale, z.T. auch formale Gruppen/Cliquen, bürokratische Organisationen oder Individuen als Akteure, die alle keine Mitglieder der Bewegung, aber miteinander vernetzte Beteiligte sind (Della Porta, Diani 1999: 16ff., 118). Dadurch können auch Unternehmen oder individuelle Software-Endnutzer als soziale Bewegungsbeteiligte eine Rolle spielen. Das untersuchte Open Source-Projekt KDE gehört zu den Open Source-Projekten, die im Zentrum der Open Source-Bewegung stehen. Jedes Open Source-Projekt hat aus Sicht der Open Source-Bewegung dabei eine spezifische Funktion inne. Das KDE-Projekt steht z.B. mit seiner grafischen Oberfläche in Beziehung zu anderen Open Source-Projekten, die darauf zurückgreifen können.

In diesem Unterkapitel werden die verschiedenen Akteure bzw. Bewegungsbeteiligten der Open Source-Bewegung wie Open Source-Projekte, Vertretungsorganisationen der Open Source-Bewegung, Infrastrukturprojekte, Unternehmen und Endnutzer vorgestellt. Dabei werden auch die Beziehungen zwischen dem untersuchten Open Source-Projekt KDE und anderen Akteuren thematisiert, um grob die Rolle und Funktion des KDE-Projekts in der Open Source-Bewegung zu umreißen. Zusätzlich soll die Open Source-Bewegung mit ihren Akteuren stärker an Kontur gewinnen und eine spätere Untersuchung der Beziehungen zwischen KDE-Projekt und Akteuren verständlicher machen. In dieser späteren Untersuchung wird anhand des Einflusses der gesellschaftlichen Projektumwelt, zu der die Open Source-Bewegung und ihre Akteure gehören, die Koordination der Softwareentwicklung im KDE-Projekt erklärt (s. Kap. 5.2.).

#### **5.1.2.1. Open Source-Projekte**

In der Open Source-Bewegung bilden Open Source-Projekte wie Linux oder das KDE-Projekt den Kern der Bewegung. Sie sind freiwillige private, nicht-staatliche Vereinigungen ohne Gewinnerzielungsabsicht und aufs Gemeinwohl orientiert, die vor allem eine gemeinsame selbstverwaltete Softwareentwicklung betreiben und ein Arbeitsnetz darstellen (s.a. Kap. 2.1.; vgl. Bentem 2001: 29f.; Brand, Holtgrewe 2007).<sup>72</sup>

---

<sup>71</sup> Beschreibungen der Lizenzen sind zu finden unter: <http://www.opensource.org/licenses/index.ph> oder Jaeger, Metzger, 2002.

<sup>72</sup> Andere Projekte werden in stärkerem Maße von Gruppen betrieben, in denen FirmenvertreterInnen als solche agieren. Sie fallen eher unter die Organisationsform überbetrieblicher F&E-Kooperationen.

Außerdem ist die produzierte Software der Open Source-Projekte, besonders wenn diese unter die GPL fällt, ein öffentliches Gut. Dieses öffentliche Gut wird von einer informalen Gruppe, die dem gruppenbezogenen Entwicklungsmodell von Ostrom (1990) ähnelt, außerhalb von Wirtschaft (Markt), öffentlicher Verwaltung (Staat) und Haushalt erstellt. Damit können alle Open Source-Projekte dem Dritten/Nonprofit-Sektor zugeordnet werden (Bentem 2001: 21ff.). Diese Zuordnung kann erfolgen, obwohl die Mitglieder der Open Source-Bewegung bzw. –Projekte meistens von ihrem Haushalt aus agieren (s.u.), da die Organisationsstrukturen außerhalb des sozialen Ortes Haushalt liegen.

Die Open Source-Projekte haben in der Open Source-Bewegung die Aufgabe, durch die Softwareentwicklung die normative Basis zum Leben zu erwecken und zu zeigen, dass die freiwillige Wissensweitergabe geschützt durch die Lizenzen funktioniert und ein Konkurrenzmodell zur proprietären Softwareerstellung darstellt. Die Projekte sind dabei als Mittel zum Zweck für einen offenen, freien Zugriff und eine Erhaltung des öffentlichen Gutes Wissen in Form von Software zu sehen.

Für eine Interaktion zwischen den Open Source-Projekten auf der einen Seite und der Außenwelt in Form von Unternehmen, staatlichen und anderen Organisationen auf der anderen Seite wurden inkorporierte formale Organisationen (Vereine oder Foundations) gegründet (vgl. O'Mahony 2003a, b; Brand, Holtgrewe 2007). Die Vereine oder Foundations haben auch die Aufgabe, Spenden von Unternehmen oder Mitgliedsbeiträge zu sammeln und z.B. beim KDE-Projekt für Werbung in Form von Messeauftritten oder KDE-Konferenzen zu sorgen oder um mittellose Mitglieder (wie Schüler) finanziell zu unterstützen, die z.B. durch den Zusammenbruch eines Computers an der Softwareentwicklung gehindert werden (Interview EB, RN). Vereine oder Foundations sollen den freien Zugriff auf den Quellcode als öffentliches Gut schützen, indem sie auf die Einhaltung der Lizenzen, wie der GPL, achten. Weiterhin kontrollieren die Vereine die Markenrechte (z.B. das KDE-Logo) oder andere Rechte ihres Projekts (vgl. O'Mahony 2003a, b). Der Verein des KDE-Projekts hat z.B. einen Vertrag mit der Firma Trolltech als Vertreiber der Basisbibliothek Qt abgeschlossen. In diesem Vertrag steht, dass bei einer Insolvenz bzw. einem Verkauf der Firma oder bei stagnierender Weiterentwicklung der Basisbibliothek Qt die Rechte an der letzten verfügbaren Version der Basisbibliothek Qt als uneingeschränkte BSD-Lizenz an den Verein fallen (Interview EB, RN).

Die Open Source-Projekte wurden aus einem Mangel heraus gegründet: bei den Bewegungsbeteiligten bestand oder besteht ein Bedarf an spezifischer Software. Daher haben sich viele verschiedene große und thematisch unterschiedliche Projekte gegründet (vgl. Brand, Holtgrewe 2007). Die Projekte bilden nach der Mitarbeiterzahl ein Kontinuum aus wenigen Großprojekten mit mehreren Hundert bis Tausend Mitarbeitern, die im Zentrum der Bewegung stehen. Extrembeispiele sind periphere, aber sehr häufig auftretende Ein-Personen-Projekte (vgl. Krishnamurthy 2002). Dazu zählen auch Softwareentwickler, die eigene Software auf Basis von Projektsoftware anderer Open Source-Projekte entwickeln, ohne Projektmitglieder zu sein (vgl. Becher et al. 2005: 120ff.).

Die Projekte sind von ihrer thematischen Ausrichtung in der Open Source-Bewegung modular ausgerichtet und ergänzen sich im Allgemeinen gegenseitig (vgl. Brand, Holtgrewe 2007). Aus Sicht der Open Source-Bewegung bearbeitet also jedes Projekt verschiedene benötigte technische Funktionalitäten bzw. Anwendungsfelder, die so auch den Markt für kommerzielle und proprietäre Software strukturieren.

Die Open Source-Software kann daher in folgende unterschiedliche Anwendungsfelder unterteilt werden:

- Software für das Internet wie Apache-Webserver, Mozilla-Internetbrowser, kryptografische Sicherheitssoftware, Firewalls, etc.,
- Software für die betriebsinterne Nutzung wie Vernetzung über Linux-Server oder Datenbanken,
- Software für die Arbeit auf (Büro-)PCs oder
- Software für die Softwareentwicklung wie die Programmiersprache Perl.

Das Produkt des KDE-Projekts gehört zur Kategorie „Software für die Arbeit auf (Büro-)PCs“, die sich aus mehreren Großprojekten für die Hauptanwendungen und vielen Kleinprojekten für Spezialanwendungen zusammensetzt, die getrennt an unterschiedlicher Software arbeiten. Die KDE- bzw. auch Gnome-Software steht als grafische Benutzeroberfläche (auch Desktop genannt) zwischen der Ebene der Betriebssysteme (z.B. Linux oder andere Unix-Derivate) und der Ebene der Anwendungssoftware (wie Open Office) als Software für die Arbeit auf (Büro-)PCs. Die einzelnen Großprojekte sind aber voneinander abhängig, um ein gemeinsames Produkt vergleichbar mit Microsoft Windows erstellen zu können, weswegen Distributionen (wie Debian) diese Software zu installierbaren Softwarepaketen bündeln (vgl. Brand, Holtgrewe 2007).

Die Open Source-Projekte können auch nach ihrer Entscheidungsstruktur unterteilt werden. Entscheidungen z.B. über Posten können in einem demokratischen Wahlverfahren wie bei Gnome (German 2003), durch einen wohlmeinenden Diktator wie bei Linux (Kuwabara 2000; Grassmuck 2002: 237f.) oder durch meritokratische Reputationsinhaber wie bei KDE (s.u.) gefällt werden. Zusätzlich gibt es einen Unterschied bzgl. der sozialen, textbasierten und technischen Offenheit gegenüber potenziellen Entwicklern.

#### *5.1.2.2. Zwei Strömungen der Open Source-Bewegung und ihre außenwirksamen Vertretungsorganisationen*

Die soziale Bewegung hat ihre Wurzeln in politisch-libertären Grundüberzeugungen, die zur Ausbildung zweier Strömungen bzw. Diskussionszusammenhänge führte. Verschiedene Lizenzen als Ausdruck des Willens für einen freien Zugang zu Wissen spielen dabei eine große Rolle. Aus diesen Strömungen bzw. Diskussionszusammenhängen wurde jeweils eine eigene formale, transnationale Vertretungsorganisation gegründet, die vor allem im Internet agiert und die Bewegung nach außen vertritt. Ihre Aufgabe in der sozialen Open Source-Bewegung besteht wie in anderen sozialen Bewegungen u.a. im Erhalt des Zusammenhalts, der politisch-normativen Diskussion der Open Source-Bewegung und dem Liefern von Identifikationsangeboten wie der Open Source Definition (s.u.) sowie der Organisation von Protesten (vgl. Della Porta, Diani 1999; Brand, Holtgrewe 2007).

Die normative Strömung, vertreten durch R. Stallman und der Free Software Foundation (FSF; [www.fsf.org](http://www.fsf.org)) als formale Vertretungsorganisation, beabsichtigt, zukünftige Nutzer und Entwickler an die Offenheit des Quellcodes zu binden. Sie entwickelten die General Public License (GPL), die am meisten von Open Source-Projekten verwendet wird und daher die wichtigste Lizenz darstellt (s.o., Weber, K. 2004; Grassmuck 2002: 230ff.; Lerner, Tirole 2002).

Die pragmatische, produktorientierte Strömung, vertreten durch E. Raymond und die Open Source Initiative (<http://www.opensource.org/>) als formale Vertretungsorganisation, sieht die Open Source-Software der proprietären, kommerziellen Software als überlegen an. Die pragmatische Strömung hat eine gemeinsame übergreifende Definition von Open Source

zur Abgrenzung von proprietärer Software (<http://www.opensource.org/docs/definition.php>) geschaffen, worunter auch die GPL fällt (s.o.). Diese Strömung ist dabei der Auffassung, dass die technische Exzellenz der Software andere gesellschaftliche Akteure von diesem Softwareentwicklungsmodell überzeugen wird und hat daher weniger Berührungsängste gegenüber Unternehmen (Weber, K. 2004; Grassmuck 2002: 230ff.).

Beide Strömungen legen großen Wert auf den Zugriff und die Weiterverbreitung von Quellcode und stehen proprietären Lizenzen eher kritisch gegenüber ([http://www.opensource.org/advocacy/shared\\_source.php](http://www.opensource.org/advocacy/shared_source.php); <http://www.fsf.org/philosophy/philosophy.html>). Bei genauerer Betrachtung divergieren beide Strömungen nur in der Sicht auf die Art und Weise der Einflussnahme, haben aber ähnlich ablehnende Ansichten hinsichtlich Nutzungsrechten oder Softwarepatenten (Weber, K. 2004: 374).

Die Lizenzen wurden von beiden Strömungen zum Schutz von Wissen bzw. Software eingeführt, damit die Open Source-Projekte mit ihrer freiwilligen Produktion öffentlicher Güter darauf aufbauen können und rechtlich abgesichert sind. Vor allem die Free Software Foundation kontrolliert die Verwendung der GPL und LGPL durch Unternehmen. Teilweise verlassen sich die Projekte auf die FSF bei der Kontrolle oder geben ihre Rechte an diese ab (O'Mahony 2003a, b; Brand, Holtgrewe 2007).

Das untersuchte KDE-Projekt unterhält Beziehungen zu beiden Bewegungen und ihren formalen Organisationen, nutzt die GPL der Free Software Foundation, ist aber eher pragmatisch auf die technische Verwirklichung ausgerichtet und weniger an politischen Auseinandersetzungen interessiert (vgl. Ettrich 2004; Brucherseifer 2004: 72).

### 5.1.2.3. Infrastrukturprojekte

Weitere Akteure in der Open Source-Bewegung sind Infrastrukturprojekte, die einerseits den Diskurs, andererseits die Softwareproduktion unterstützen.

Zu den Infrastrukturprojekten gehören also offen zugängliche Internet-Kommunikationsplattformen (Newsportale), die für die Open Source-Bewegung und -Projekte eine Infrastruktur für die Kommunikation zur Verfügung stellen (Brand, Holtgrewe 2007). Damit wird der politisch-normative Diskurs und die Abstimmung zwischen den verschiedenen Open Source-Projekten unterstützt. Die Projekte und die formalen Vertretungsorganisationen der Bewegung informieren auf den Kommunikationsplattformen z.B. über neuere Entwicklungen. Darüber werden Ankündigung und Erscheinen neuer offizieller, stabiler KDE-Software-Versionen bekannt gemacht. Dabei steht die Aufmerksamkeits-erzeugung im Vordergrund, die auch der Sanktionierung von Unternehmen dient, die die GPL-Lizenz nicht beachten (Holtgrewe, Brand 2007). Davon hat auch das KDE-Projekt in mindestens einem Fall profitiert. Durch die Thematisierung erhält das Unternehmen einen Imageschaden (vgl. O'Mahony 2003a, b; Holtgrewe, Brand 2007). Der politisch-normative Diskurs findet im Allgemeinen auf News-/Kommunikationsplattformen wie slashdot.org, newsforge.org, etc. oder auf den Seiten der formalen Vertretungsorganisationen der Bewegung statt.

Zu den Infrastrukturprojekten gehören des Weiteren Plattformen für die virtuelle Zusammenarbeit, d.h. Anbieter von internetbasierten Versionsmanagementsystemen wie sourceforge.org. Dabei wird z.B. sourceforge.org eher von kleinen Projekten genutzt, die keine eigenen Werkzeuge schreiben oder anpassen können, da zu wenige Mitarbeiter mitentwickeln (Krishnamurthy 2002). Die großen Projekte dagegen nutzen oder entwickeln eigene Werkzeuge wie Website/Homepage, Versionsmanagementsystem, etc. Das KDE-Projekt

hatte von Anfang an eine eigene IuK-technische Infrastruktur und war nicht auf solche Plattformen angewiesen.

#### 5.1.2.4. Open Source-Bewegung und Unternehmen

Unternehmen treten als weitere Akteure in der sozialen Bewegung und im Umfeld der Projekte auf, wobei Beziehungen zwischen den Unternehmen und der Bewegung bzw. den Projekten bestehen. Die Open Source-Bewegung bzw. –Projekte lehnen die wirtschaftliche Sphäre bzw. die Unternehmen als deren Akteure nicht von vorneherein ab (s.o.). Die Unternehmen können eher nach dem Grad der Beziehungsstärke oder Interessenkongruenz als Beteiligte bis hin zu einer Gegnerschaft gegenüber der Bewegung eingeteilt werden.<sup>73</sup>

Manche Unternehmen können als Beteiligte der Open Source-Bewegung gesehen werden, da sie einerseits die soziale Bewegung unterstützen und einen Großteil ihrer Ansichten teilen. Zusätzlich veröffentlichen sie Software unter der GPL-Softwarelizenz und unterwerfen sich den Werten und Normen der sozialen Bewegung. Dies ist nicht ungewöhnlich, da aus und in anderen sozialen Bewegungen wie der Friedens- und Umweltbewegung Genossenschaften, selbstverwaltete Betriebe und Bioläden entstanden und den Regeln und Ansichten der Bewegung folgten (vgl. Beywl 1987; Atmaca 2002; Heider, Mevissen 1991).<sup>74</sup>

Zusätzlich sind Unternehmensmitarbeiter als Open Source-Mitarbeiter bei der Softwareentwicklung beteiligt (s.u.). Manche Unternehmen unterstützen die Ausbreitung der Softwareprodukte und –lizenzen der Bewegung. Andererseits gehören die Unternehmen nicht vollständig zur Bewegung und damit an ihren Rand, da sie zwar überschneidende, aber auch andere, gewinnorientierte, betriebswirtschaftliche Interessen als die der sozialen Bewegung verfolgen.

Da in der Bewegung die Softwareproduktion und nicht eine vollkommene Ablehnung gegen Unternehmen im Vordergrund steht, steht einer Zusammenarbeit mit Unternehmen unter bestimmten Bedingungen nichts im Wege. Unternehmen gehören also als Beteiligte dann zur Bewegung, wenn sie den Regeln, Lizenzen und Ansichten der sozialen Bewegung folgen und überwiegend aus Personal bestehen, dass der Open Source-Bewegung nahe steht.

Die Unternehmen, die an der Open Source-Bewegung beteiligt sind, sind einerseits sogenannte Start-Ups, die sich wie die selbstverwalteten Betriebe aus der sozialen Bewegung entwickelt haben, aber auch größere etablierte Softwareunternehmen (vgl. Gruber, Henkel 2004). Zwischen diesen Unternehmen und Open Source-Projekten wie dem KDE-Projekt besteht eine gegenseitige Abhängigkeit, was sich in reziproken Beziehungen darstellt.<sup>75</sup> Das KDE-Projekt überlässt den Unternehmen das Softwareprodukt zur Weiterverbreitung und ermöglicht den Unternehmen so ihr Dienstleistungsgeschäftsmodell, d.h. das Anbieten von Dienstleistungen rund um das Produkt. Diese Businessmodelle konzentrieren sich z.B. um das Anpassen oder Neuschreiben von Open Source-Software bei eher kleinen Servicefirmen bzw. Start-Ups. Weitere Firmen schnüren und verkaufen Pakete aus Linux-

---

<sup>73</sup> Ein gewisses Maß an Misstrauen von Open Source-Projekten gegenüber der wirtschaftlichen Sphäre zeigt sich z.B. durch die Absicherung über formale Organisationen und Markenrechte (O'Mahoney 2003a, b; Brand, Schmid 2003a; Brand, Holtgrewe 2007).

<sup>74</sup> Einige haben sich aber nach einiger Zeit den Wirtschaftsgewohnheiten angepasst, was auch bei den Open Source-nahen Unternehmen geschehen kann, wobei eine zahlenmäßige Verringerung der projektnahen Unternehmen zu erwarten ist (vgl. Beywl 1987; Atmaca 2002; Heider, Mevissen 1991).

<sup>75</sup> Die gegenseitige Abhängigkeit kann je nach Projekt unterschiedlich stark sein und bedarf weiterer Untersuchungen.

Betriebssystem, Desktops und Anwendungsprogrammen und z.T. proprietären Installationsprogrammen (vgl. Pal, Madanmohan 2002; Leiteritz 2002; Brand, Holtgrewe 2007). Damit hängen diese Unternehmen von der Softwareentwicklung der Open Source-Projekte ab.

Im Gegenzug geben die Firmen den Open Source-Projekten, seltener auch den Vertretungsorganisationen, Spenden für die Organisation von Konferenzen, für Auftritte auf Linux-Konferenzen und stellen ihre Server für Website, Mailinglisten und Dateiablagensystem der Projekte zur Verfügung. Außerdem stellen Unternehmen Softwareentwickler wie aus dem KDE-Projekt ein, die auch während ihrer Arbeitszeit an der Open Source-Software mitarbeiten. Wenn Unternehmen den gleichen Quellcode nutzen, geben sie Fehlerberichtigungen zurück, wobei alle anderen Unternehmen auch von diesen profitieren (vgl. Working Group 2000: 15f.; Brand, Holtgrewe 2007). Im Falle des KDE-Projekts besteht eine solche kooperative Beziehung mit der Firma Trolltech. Die Firma verlangt einerseits Lizenzgebühren für diese proprietäre Bibliothek, andererseits wird die Bibliothek unter der GPL als KDE-Basisbibliothek genutzt.<sup>76</sup>

Es gibt auch Unternehmen, die die Open Source-Software als Mittel zum Zweck verwenden, aber nicht mit den Ansichten der Open Source-Bewegung konform gehen. Manche Softwareunternehmen verkaufen ihre Hardware-Produkte mit Open Source-Software als Zugabe (z.B. IBM). Andere z.T. größere Unternehmen nutzen Open Source-Software als Druckmittel gegen den Monopolisten Microsoft oder andere mächtige Unternehmen (s. z.B. Netscape mit Mozilla, Sun mit Open Office/StarOffice<sup>77</sup>).

Weiterhin gibt es Gegner der Open Source-Bewegung bzw. von Projekten, die die normative Ausrichtung der Bewegung ablehnen. Dazu gehört z.B. die Firma Microsoft, die eine Bedrohung ihres hohen Marktanteils durch Open Source-Software sieht und mehr oder weniger offen gegen die Open Source-Bewegung agiert (s. Halloween Paper, SCO-Unterstützung<sup>78</sup>). Im Gegenzug wird die Firma Microsoft aufgrund ihres Geschäftsgebahrens und ihres Marktanteils von der Open Source-Bewegung beobachtet.

#### 5.1.2.5. *Open Source-Bewegung und individuelle Beteiligte*

Die Open Source-Bewegung besteht wie andere soziale Bewegungen aus informalen z.T. auch formalen Cliquen, Gruppen und bürokratischen Organisationen, die sich aus individuellen Beteiligten zusammensetzen (vgl. Della Porta, Diani 1999). Die miteinander vernetzten Individuen der Open Source-Bewegung sind vor allem junge hochqualifizierte Männer zwischen 20 und 30 Jahren (vgl. Ghosh et al. 2002; Lakhani, Wolf 2003). Diese jungen Menschen haben durch ihre Berufswahl – Informatik oder ähnliche Disziplinen – eine Affinität zum Konflikt, auf den die soziale Bewegung aufmerksam macht bzw. zu den Überzeugungen der Open Source-Bewegung. Damit unterscheidet sich die Open Source-Bewegung nur durch das vorherrschende Auftreten der Männer und der „Informatiklastigkeit“ von den anderen sozialen Bewegungen, wie Umwelt-, Friedens- und Anti-Globalisierungsbewegung, die wie die Open Source-Bewegung vor allem durch junge Menschen der Mittelschicht getragen werden (vgl. Hellmann 1998: 16; Rucht et al. 1997:

---

<sup>76</sup> Doppellizenzierungen sind möglich, dabei kann eine Software unter der GPL nicht proprietär angeeignet werden (vgl. Välimäki 2003; Jaeger, Metzger 2002: 79ff.). Die Firmen können dadurch das Informationsparadoxon überwinden, d.h. bei einem Verkauf von Informationen ist der Preis einer Information von ihrer Nicht-Bekanntheit abhängig. Für bekannte Informationen kann kein Preis erzielt werden (vgl. Franck, Jungwirth, 2002a).

<sup>77</sup> s.a. <http://de.wikipedia.org/wiki/Mozilla-Projekt> und <http://de.wikipedia.org/wiki/OpenOffice.org>

<sup>78</sup> s. <http://www.opensource.org/halloween/>; <http://www.heise.de/newsticker/meldung/45237>; Achtenhagen et al. 2003; Grassmuck 2002: 206, 299; Leiteritz 2002; 2003.

103; Brand 1998: 37). Die thematisch nahestehenden Berufe können die Männerdominanz und die Dominanz der Informatik erklären. Es ist zu vermuten, dass die geringe Anzahl an Frauen im Informatikstudium den geringen Anteil an Frauen in der Open Source-Bewegung und im –Projekt begründet. Die mangelnde Internetnutzung bestimmter Bevölkerungsgruppen in den 1990er Jahren könnte ein weiterer Grund für die Begrenzung auf männliche, junge Hochqualifizierte sein, wobei diese geringe Internetnutzung kaum mehr eine Rolle spielt ((N)onliner Atlas 2004: 12ff.).

Viele dieser Beteiligten sind in Open Source-Projekte, die Vertretungsorganisationen und Infrastrukturprojekte involviert. Da sie häufig mitarbeiten, können sie identifiziert und angesprochen werden. In der Open Source-Bewegung gibt es aber auch unorganisierte, lose gekoppelte Beteiligte (Sympathisanten), die entweder auf Basis von Open Source-Software eigene Software entwickeln oder nur die Software der Open Source-Projekte als Endnutzer nutzen. Die Endnutzer als Umwelt der Projekte und z.T. Beteiligte der sozialen Bewegung können in mehrere Bereiche eingeteilt werden. Die einen sind ständige und aktive Nutzer der Software, die sich an der Entwicklung mit Fehlerberichten beteiligen. Sie senden sehr oft Wünsche und Fehlerberichte an die Softwareentwickler z.B. des Open Source-Projekts KDE. Diese können wiederum in Nutzer mit einem hohen oder niedrigen Maß an Programmierkenntnissen unterteilt werden. Ein Großteil ist aber passiv, d.h. die Endnutzer beteiligen sich entweder gar nicht oder nur einmal. Im Weiteren wird auf individuelle Endnutzer fokussiert.

## **5.2. Akteure des Open Source-Projekts KDE**

Allgemein ist die Open Source-Bewegung auf den normativen Diskurs, den Protest und die Einhaltung der Lizenz ausgerichtet, wogegen die Open Source-Projekte sich auf die Produktion der Software konzentrieren. Dabei ist das Open Source-Projekt KDE, das im Folgenden als Untersuchungsobjekt im Fokus steht, eine wichtige Komponente der Open Source-Bewegung. Das KDE-Projekt produziert eine grafische Benutzeroberfläche für Betriebssysteme vorzugsweise für den Bereich der Linux- bzw. Unix-Derivate, aber auch diverse Anwendungssoftware. Die entwickelte Software ist dabei weitestgehend unter der GPL oder der LGPL lizenziert.

Der Beginn der Softwareentwicklung ist auf Ende 1996 zu datieren. Drei Studenten gründeten das KDE-Projekt an der Universität Tübingen, das schnell auf eine Gruppe von ca. 20 Personen anwuchs. Mittlerweile sind ca. 1000 Mitarbeiter zu verzeichnen. Diese Mitarbeiter sind mit der Koordination des Arbeitsprozesses der Softwareentwicklung im KDE-Projekt verbunden, wobei einige Mitarbeiter einen stärkeren Einfluss auf die Koordination haben als andere.

Im Folgenden werden verschiedene Charakteristika der Mitarbeiter als individuelle Akteure dargestellt. Im Unterkapitel 5.2.1. wird auf die sozioökonomischen Charakteristika der individuellen Akteure und im Unterkapitel 5.2.2. auf ihre Motivation eingegangen, wobei sich die KDE-Mitarbeiter weitestgehend kaum von anderen Open Source-Projekten bzw. von der -Bewegung unterscheiden. Im Unterkapitel 5.2.3. werden die Tätigkeits- und Strukturbereiche vorgestellt, in denen die individuellen Akteure arbeiten. Basis der Darstellung sind neben den Interviews besonders die quantitativ-standardisierten Befragungen auf einer Konferenz und die Selbstauskünfte-Interviews (s. Anhang B, C, D). Ausgewählte Daten werden im Anhang dargestellt.

### **5.2.1. Sozioökonomische Charakteristika**

Die Akteure des Open Source-Projekts KDE sind wie in anderen Open Source-Projekten vor allem autonome Privatpersonen, Unternehmen oder Verbände. Eine Organisations-

grenze durch Mitgliedschaft im Projekt ist heute schwer zu ziehen, d.h. der Übergang zwischen Innen und Außen des Projekts ist diffus. Im Fehlerberichtssystem wurden nach Angaben seines Administrators vom 1. bis 31. Oktober 2004 1050 eingesandte Fehlerberichte verzeichnet, wobei auch Personen von außerhalb des Projekts Fehler melden. Schreibzugriff auf das Versionsmanagementsystem, das den bearbeiteten Quellcode, Übersetzungen, Dokumentationen etc. beheimatet (s.u.), haben dagegen Ende 2004 nur 915 KDE-Mitglieder (vgl. Robles 2004). Auf Tagungen des Projekts kommen aber nur 100-200 Personen zusammen, doch die Anzahl der Anwesenden steigt weiterhin an.<sup>79</sup>

Als offizielle oder eingeschriebene Mitglieder werden Personen definiert, die Schreibzugriff auf die IuK-technische Infrastruktur des Projekts haben, d.h. auf das Versionsmanagementsystem (s.u.). Als Dunstkreis oder quasi-Mitglieder ohne offiziellen Status sind alle Personen zu verstehen, die wiederholt, über einen gewissen Zeitraum hinweg (z.B. über ein halbes Jahr), mindestens kleinere Arbeiten (Fehlerberichte, wenige fehlerberichtigende Quellcodezeilen, Organisation von Konferenzen, etc.) durchführen und sich zeitweise auch an der Kommunikation beteiligen (für eine Einschätzung aus KDE: s. Brucherseifer 2004). Dabei legen die offiziellen Mitglieder die Arbeitspakete der inoffiziellen Mitarbeiter für diese ins Versionsmanagementsystem. Insgesamt können die offiziellen und inoffiziellen KDE-Mitarbeiter auf 1000 bis 1200 Personen geschätzt werden.

Die Mitglieder sind weltweit verteilt und arbeiten über das Internet zusammen. Eine räumliche Häufung der Mitglieder ist in Europa, besonders in Deutschland, wahrscheinlich wegen des Gründungsorts, zu finden (s. Anhang A Tab. A 1). Die Mitglieder sind demografisch homogen: Eine Majorität ist männlich und zwischen 20 und 30 Jahre alt. Sie haben einen IuK- bzw. IuK-nahen Hochschulabschluss oder streben diesen an (s. Anhang A Tab. A 2, Anhang A Tab. A 3, Anhang A Tab. A 4).<sup>80</sup> Damit unterscheiden sich die KDE-Mitglieder kaum von anderen Open Source-Beteiligten (vgl. Robles et al. 2001; Ghosh et al. 2002; Lakhani, Wolf 2003; Hertel et al. 2003; für eine Zusammenschau: Holtgrewe 2004b: 344f.).

Nur eine sehr kleine Minderheit wird für die Arbeit an dem KDE-Projekt wie bei anderen Projekten bezahlt, d.h. die Mehrheit arbeitet freiwillig, ohne Lohn an Open Source-Projekten. Ein paar können aber gelegentlich Geld damit verdienen bzw. arbeiten neben ihrer normalen Arbeitszeit für das Projekt. Drei Selbstaussagen belegen dies:

1. „Since the beginning of August I'm paid by [Firmenname] to work on [the project] fulltime. It's really great. I used to have another job during the day and work on [the project] during the nights, but given how much I'm involved into KDE, it was just too much. I really enjoy being able to work fulltime on [the project] now!“ (Selbstaussage 19).

2. „I'm offering my services for holding tutorials and courses on [Programmname] for companies though, as well as earning some money with writing articles about [Programmname] for computer magazines“ (Selbstaussage 48).

3. „[Program] is a great achievement because none of the people working on [program] is payed to work on it. Some companies let sometimes one of its employees work on it, or subsidies work on a small part of it. (For example my company, [firm name], let me sometimes work on it.) But most of the work on [program] is done as a hobby by the developers or the users and [program] can still compete very well with all the commercial IDE“ (Selbstaussage 41).

---

<sup>79</sup> Dies liegt an der weltweiten Verteilung und den hohen Reisekosten.

<sup>80</sup> Bei der quantitativen Befragung von 55 KDE-Mitgliedern hatten 28 Personen Informatik, 13 Personen Mathematik, Ingenieur- oder Naturwissenschaften und 4 Personen diverse andere Ausbildungen bzw. Studiengänge absolviert bzw. absolvieren diese gerade.

Es werden z.B. für das KDE-Projekt zwischen 5 und 10 Entwickler Vollzeit nach Aussage eines Interviewten von einer Firma für die Arbeit an KDE bezahlt. Zusätzlich können Firmen-Softwareentwickler während ihrer Arbeitszeit an der KDE-Software mitarbeiten (vgl. <http://www.kde.org/support/sponsoring/thanks.htm>; Robles et al. 2001; Ghosh et al. 2002; Holtgrewe 2004b: 345).

Die Zusammenarbeit im Projekt wird als ein Hobby betrachtet, wobei eine Bastlermentalität dies unterstützt. Ein Beispiel für Bastlermentalität zeigt die Selbstaussage 60: „When I see something new, I want to know everything about it. I especially like new things in technology” (Selbstaussage 60). Dabei muss dieses Hobby mit Familie, Erwerbsarbeit und anderen Hobbys konkurrieren. Die Balance zwischen KDE-Projekt und anderen Tätigkeiten verdeutlichen die beiden folgenden Selbstaussagen:

„As much as I can, I have to keep a good balance between KDE and my family” (Selbstaussage 28) und „I try to do as many activities that are not related to computers in my spare time as possible. This includes lots of sports like inline skating, going to the movies, meeting friends and going out” (Selbstaussage 12).

Die meisten Austritte geschehen durch eine Veränderung der Prioritäten in der Freizeit, so dass die Arbeit im Projekt überwiegend als Lebenslaufphänomen der verlängerten Adoleszenz gesehen werden kann. Dies zeigt sich auch in der Altersstruktur von Anfang bis Ende Zwanzig Jahren (s.o.; vgl. Brand, Holtgrewe 2007). Das Lebenslaufphänomen wird durch folgende Selbstaussage plastischer: „It seems as if university graduation is KDE's biggest enemy in terms of where we lose our developers”. Die Mitarbeiter ziehen sich oftmals unbemerkt zurück und treten nicht mehr in Erscheinung. Dadurch können keine weiteren Aussagen zum Austritt aus dem Projekt gemacht werden.

### **5.2.2. Motivation**

Die Motivation der individuellen Beteiligten der Open Source-Bewegung bzw. der -Projekte wurde vor allem aus Sicht verschiedener Autoren als rationale, sich am Eigeninteresse ausrichtende Akteure beschrieben (s. z.B. Lerner, Tirole 2000; Ghosh 1998; Franck, Jungwirth 2002a, b; Weber 2000).<sup>81</sup> Nach Raymond (1998a, b), einem wichtigen Sprecher der Open Source-Bewegung, dominieren dagegen eher nicht-egoistische Motive<sup>82</sup>, was von den Ergebnissen verschiedener Online-Befragungen bestätigt wird (vgl. Robles et al. 2001; Ghosh et al. 2002; Lakhani, Wolf 2003; Hertel et al. 2003). Mittlerweile wird von dominierenden intrinsischen neben extrinsischen Motivationen gesprochen (vgl. z.B. Osterloh 2002; Luthiger 2004; Holtgrewe 2004b).<sup>83</sup>

Die Motivationen der KDE-Projektmitglieder unterscheiden sich wenig von den Ergebnissen der Online-Befragungen von anderen Projekten (vgl. s.o.; Holtgrewe 2004b: 345f.). Die Befragungen von KDE-Projektmitgliedern und Mitgliedern anderer Projekte ergaben, dass diese hochgradig intrinsisch motiviert sind und extrinsische Motivationsgründe eine untergeordnete, monetäre Anreize kaum eine Rolle spielen. Der Spaß am selbstbestimmten Programmieren von z.T. störenden Problemen der Software für den Eigenbedarf, Wei-

---

<sup>81</sup> Die Aussagen „But on the Net, a cooking-pot market is far from altruistic, or it wouldn't work” von Ghosh (1998) oder “Why should thousands of top-notch programmers contribute freely to the provision of a public good? Any explanation based on altruism only goes so far” von Lerner/Tirole (2000) sind dafür typisch.

<sup>82</sup> „It consciously distrusts and despises egotism and ego-based motivations; self-promotion tends to be mercilessly criticized, even when the community might appear to have something to gain from it” (Raymond 1998b).

<sup>83</sup> Intrinsische Motivation bezieht sich aus der Bedürfnisbefriedigung über die Tätigkeit bzw. Beziehung selbst, extrinsische auf die Bedürfnisbefriedigung durch außerhalb der Person stehende Anreize (vgl. Osterloh 2002).

terentwickeln der eigenen Fähigkeiten und natürlich auch der Stolz auf die geleistete Arbeit stehen in der quantitativen Befragung im Vordergrund.<sup>84</sup> Dabei ist die selbstbestimmte Softwareentwicklung im KDE-Projekt ein Ausgleich für anderweitige Arbeit wie der eher fremdbestimmten Arbeit in Unternehmen. Dies wird durch die Selbstaussage (Nr. 54) „Coding is holiday, fun and recreation for me“ verdeutlicht.

Weiterhin ist das Erringen von sozialer Anerkennung bzw. Reputation ein extrinsischer Motivationsgrund, der nicht direkt, sondern nur als Nebenfolge angestrebt werden kann. Reputation kann als symbolisches Kapital aufgefasst werden (vgl. Zeitlyn 2003), das die Projektbeteiligten sich verdienen müssen. In der Befragung der KDE-Mitglieder rangierte diese Motivation am unteren Ende der Wichtigkeitsskala, spielte aber in den Interviews eine bedeutendere Rolle. Im KDE-Projekt liegt also auch der Anreiz darin, dass die Einbindung und Position im Projekt auch von den ständigen Arbeitsleistungen mit qualitativ hochwertigem, d.h. fehlerfreiem Quellcode und der ständigen sichtbaren Beteiligung auch in Email-Diskussionen abhängt. Der qualitativ hochwertige Quellcode ist ein Ausdruck für Erfahrung, was sich auch in hochwertigen bzw. durchdachten Diskussionsbeiträgen niederschlägt. Außerdem geht Reputation mit einem kooperativen Kommunikationsstil einher: Prahlen ist verpönt (vgl. Brand, Holtgrewe 2007).

Aber auch Teamarbeit, Ablehnung von Microsoft bzw. proprietärer Software, Reziprozität sowie Verpflichtung gegenüber den Projektmitgliedern oder bessere mögliche Einstiegschancen in das Berufsleben spielen eine, wenn auch geringere Rolle.

Die Norm der Verpflichtung bzw. der Reziprozität zeigt sich in Selbstaussage 18: „I was so impressed with the work of such a high quality that others did for nothing, that I decided to respond with same kind of work, both for KDE and Linux.“ Die Ablehnung von Microsoft wurde in folgender Selbstaussage 37 ausgedrückt: „One of the stronger open source projects, proving Microsoft wrong for thinking only they can deliver desktop products and proving Apple wrong (sorry Jordan Hubbard) for thinking only a company like them could combine UNIX with the desktop“.

So gaben z.B. zwei Projektbeteiligte in den Interview-Selbstauskünften (Nr. 60 und 4) auf die Frage, ob sie angestellt sind, an: „No, but I wouldn't mind :o)“ und „No. [\* employers, make mental note]“. Diese Aussagen zeigen, dass die Projektbeteiligten das Hobby gerne zum Beruf machen würden, aber nicht damit rechnen. Einen finanziellen Vorteil verspricht sich keiner.

Die Dominanz der intrinsischen Motivation bedeutet nicht, dass wenig gearbeitet würde. Im Gegenteil: Ins Versionsmanagementsystem werden pro Tag im Median 1155,73 Beiträge gestellt, dies sind im Median 458 Beiträge pro Person pro Jahr (Robles 2004). Die Arbeitszeit schwankt nach der quantitativen Befragung zwischen 0,5 und 90 Stunden pro Woche (Median 10 Std.) und den Selbstaussagen zwischen 0,14 und 14 Stunden am Tag (Median 2,9 Stunden pro Tag). So bekannten zwei KDE-Mitglieder in Selbstaussagen (Nr. 7 und 37): „Some days several hours, other days none“ bzw. „That depends a lot, some weeks it's hardly any work at all; other weeks it's several hours per evening.“ Gründe sind nach den Selbstaussagen die schwankenden Anforderungen während der Softwareentwicklung mit arbeitsintensiven Phasen wie dem Release, aber auch die schwankenden projektexternen Anforderungen wie Erwerbsarbeit, Studium, Familie etc. Zusätzlich ist nach der quantitativen Auswertung der Arbeitseinsatz von wenigen zentralen höher als bei eher peripheren KDE-Mitarbeitern (s.u.). Eine Minderheit arbeitet fast Vollzeit, eine Mehrheit nur wenige Stunden in der Woche an dem Projekt, was auch in anderen Projekten der Fall ist (vgl. Holtgrewe 2004b: 345).

---

<sup>84</sup> Die intrinsische Motivation zeigt sich auch bei 24 von 53, also knapp der Hälfte, der quantitativ befragten KDE-Mitglieder, die einer ehrenamtlichen Tätigkeit nachgehen oder nachgegangen sind.

### 5.2.3. Tätigkeits- und Strukturbereiche

Die KDE-Mitglieder als individuelle Akteure arbeiten freiwillig in verschiedenen Tätigkeits- und Strukturbereichen des KDE-Projekts. Das Zentrum-Peripherie-Prinzip prägt diese Tätigkeits- und Strukturbereiche des Open Source-Projekts KDE (Brand, Holtgrewe 2007). Diese Zentrum-Peripherie-Struktur zeigt sich, wie z.B. oben betrachtet, in der geografischen Verteilung der KDE-Mitarbeiter. Dies gilt auch für die verschiedenen Tätigkeitsbereiche, wie Softwareentwicklung, technische Dokumentation/Übersetzung, Grafik-/Sound-/Design-Entwicklung, Fehlerberichterstattung, Vereinsarbeit, Konferenzorganisation/Werbung, Infrastrukturüberwachung etc., die sich u.a. jeweils in eigenen Homepages und Mailinglisten wiederfinden. Die Softwareentwickler haben einen zentralen Stellenwert im Projekt, die Dokumentare/Übersetzer sind diesen nachgeordnet und weitere Tätigkeiten haben eine periphere Bedeutung. Diese periphere Bedeutung zeigt sich in folgender Selbstaussage 25: „I really admire what the KDE translators are doing. They are the silent heroes of KDE, translating and translating again, whatever we thoughtless developers are committing“. Dabei stellen die Softwareentwickler eine Mehrheit von ungefähr zwei Dritteln aller Mitarbeiter gegenüber den restlichen Tätigkeiten mit einem Drittel aller Mitarbeiter (Interview DF). Technische Dokumentare/Übersetzer sind die zweitgrößte Gruppe. Nur eine geringe Anzahl bearbeitet die anderen Tätigkeiten.

Der Übergang zwischen den Tätigkeiten Softwareentwicklung, Dokumentation/Übersetzung und den restlichen Tätigkeitsbereichen ist aber fließend, da z.B. Softwareentwickler übersetzen/dokumentieren und umgekehrt (s.a. Anhang A Tab. A 5). Die meisten Übersetzer, Dokumentare oder Designer sind eher fachfremd, doch werden sie manchmal zu Entwicklern, so dass sie einen Pool für neue Mitentwickler darstellen.

Bei den Softwareentwicklern existiert wiederum ein Zentrum und eine Peripherie, die als „Reputationshierarchie“ bzw. „meritokratische Oligarchie“ bezeichnet werden können: Es gibt zehn Softwareentwickler von 740 Entwicklern Anfang 2003, die ca. 45 % der Quellcodezeilen verändert haben bzw. für ca. 54 % der Versionsmanagementzugriffe verantwortlich sind (s. Anhang A Abb. A 1). Dies zeigte sich auch in der Verteilung der Arbeitszeit (s.o.).

Dabei führt die Arbeit selbst nicht automatisch zu einer hohen Anerkennung, sondern sollte auch für andere sichtbar sein, was in Selbstaussage Nr. 32 belegt wird:

„The people who manage the project's infrastructure (servers, mailing lists, build tools etc), stuff that users and developers never directly see but depend on so heavily. It's not rock star work, so you can't doubt the commitment of folks like that.“

In dieses Zentrum der Softwareentwickler gehören auch die Gründer des KDE-Projekts, die vorher in dem Open Source-Projekt Lyx ([www.lyx.org](http://www.lyx.org))<sup>85</sup> arbeiteten und weitere Mitarbeiter aus diesem Projekt, aber auch aus anderen Linux-Projekten zur Mitarbeit aufforderten.

Der Stellenwert der vielen Subprojekte, in denen die Softwareerstellung im KDE-Gesamtgroßprojekt stattfindet, unterscheidet sich auch nach Zentrum und Peripherie. Dabei ist durch die Modularität der Software jedes Subprojekt für einen spezifischen Softwarebereich bzw. ein Programm(-modul) im Versionsmanagementsystem (s.u.) zuständig. Es gibt Subprojekte mit zentraler, wichtiger Software für das restliche Projekt, wie die Kernbibliotheken mit oft genutzten Funktionalitäten, auf die andere Programme zurückgreifen. Peripher sind Subprojekte, in denen zusätzliche, aber nicht notwendige Anwendungs-

---

<sup>85</sup> Im Lyx-Projekt wird eine einfache grafische Oberfläche für das komplizierte Desktop Publishing-Programm Latex entwickelt, das über eine Befehlszeile programmiert wird und auch unter der GPL steht.

programme wie Witzprogramme, Spiele etc. erstellt werden (s.u.). Zudem umfassen diese wenigen zentralen Subprojekte im KDE-Gesamtprojekt um die 20 bzw. 30 Personen. Im Durchschnitt bestehen die Subprojekte aus vier Personen, es gibt jedoch auch viele periphere Ein-Personen-Subprojekte (vgl. Anhang A Abb. A 2). Innerhalb der Subprojekte besteht wiederum eine konzentrische Struktur: Es gibt jeweils einen für das Projekt verantwortlichen Leiter und weitere Mitglieder. Dabei ist die Bedeutsamkeit eines Subprojektleiters als Posten für das Gesamtprojekt von der Bedeutsamkeit des Subprojekts abhängig.

Neben diesen unterschiedlich wichtigen Subprojektleiterposten gibt es noch andere legitimierte zentrale und periphere Posten im Projekt. Die zentralen Posten haben Aufgaben, die das gesamte Projekt beeinflussen, wie Infrastrukturpflege mit technischer Schreibzugriffsvergabe oder Releasekoordination (Beschreibung des Release s. Kap. 2.2.). Periphere Posten beziehen sich nur auf Teilbereiche des Projekts wie Koordinierung des KDE-Edutainment-Bereichs mit Lernprogrammen oder des journalistischen Newstickers auf der KDE-Homepage. Weitere formale Posten mit nachgeordnetem Stellenwert bei der Softwareentwicklung haben sich zusätzlich im KDE-Projekt wie bei anderen Open Source-Projekten (vgl. O'Mahony 2003a, b) im Verein KDE e.V. als formaler Organisation gebildet.

Diese konzentrische Struktur ist auf eine horizontale und vertikale Arbeitsteilung zurückzuführen. Hierarchische Elemente in Form von legitimierten Posten treten neben Reputation auf.

### **5.3. IuK-technische Infrastruktur des Open Source-Projekts KDE**

Eine Besonderheit der Open Source-Projekte ist die fast vollständige Virtualität der Arbeit, d.h. die alltägliche Produktion und Kommunikation läuft nur über die Informations- und Kommunikationstechnik (IuK-Technik) ab. Die hohe Virtualität ist die Basis für das Entstehen und Bestehen des KDE-Projekts, da die hohe geografische Streuung gleichinteressierter (potenzieller) Mitarbeiter persönliche Treffen überwiegend verhindert und diese auf seltene Gelegenheiten wie organisierte Projekt- oder Linux-Konferenzen verringert. Es gibt aber auch regionale Cluster oder Mikrogruppen von Projektbeteiligten, die am gleichen Ort bzw. in der gleichen Region oder Stadt wohnen und sich deswegen öfter persönlich treffen. Dies gilt vor allem für Deutschland.

Da die Virtualität im KDE-Projekt einen hohen Stellenwert hat, wird in diesem Unterkapitel 5.3. die IuK-technische Infrastruktur vorgestellt, die die Basis für die Softwareentwicklung im Open Source-Projekt KDE darstellt. Dabei ist (s. Kap. 2.2.) zwischen dem untersuchten Endprodukt und der unterstützenden IuK-technischen Infrastruktur als Technik zu unterscheiden. Im KDE-Projekt werden als End-/Gesamtprodukt neben der Hauptsoftware einer grafischen Benutzeroberfläche (Desktop) auch weitere Anwendungsprogramme wie Spiele, Edutainment-Programme, Office-Programme, Unterstützungssoftware für Programmierung/Übersetzung (z.B. Dokumentations-/Übersetzungs-/Quellcodeeditoren, ein Debugger, eine grafische Versionsmanagement-Navigationsoberfläche)<sup>86</sup> etc. entwickelt. Grundlage dieser Programme ist seit Beginn des Projekts die frei zugängliche, technisch hochentwickelte Qt-Bibliothek der Firma Trolltech. Wie oben beschrieben, ist das Softwareprodukt des KDE-Projekts ein öffentliches Gut, die IuK-technische Infrastruktur kann dagegen als ein Clubgut gesehen werden, bei dem nur die Clubmitglieder, d.h. die KDE-Mitglieder, Zugriff haben. Der Lese- und Schreibzugriff ist für die einzelnen Werkzeuge der IuK-technischen Infrastruktur in abgestufter Form möglich, d.h. es können entweder alle, nur eine eingeschränkte Gruppe oder nur wenige der KDE-Mitarbeiter auf diese Infrastruktur zugreifen. Die IuK-technische Infrastruktur wird als gekapseltes Artefakt gesehen, das

---

<sup>86</sup> s. das KDE-Subprojekt Kdevelop: [http://www.kdevelop.org/index.html?filename=links\\_tools.html](http://www.kdevelop.org/index.html?filename=links_tools.html).

im Sinne Heideggers ein Gestell, im Sinne der „Actor-Network Theory“ eine Black Box oder ein Zweck-Mittel-Verhältnis ist (s. Kap. 2.2.). Die bisherige stabile IuK-technische Infrastruktur bleibt weitestgehend unverändert, während andere Software weiterentwickelt wird.<sup>87</sup> Eine fehleranfällige IuK-technische Infrastruktur würde sonst eine Zusammenarbeit verhindern.

Werden die Endprodukte der grafischen Benutzeroberfläche und die Unterstützungssoftware für Programmierung/Übersetzung direkt auf sich selbst angewendet, ist dies bei der IuK-technischen Infrastruktur und anderer Software nicht der Fall. Bei den Endprodukten der grafischen Benutzeroberfläche und der Unterstützungssoftware werden die neuesten instabilen Entwicklungsversionen während ihrer Softwareentwicklung genutzt und Erfahrungen mit diesen direkt in die weitere Softwareentwicklung integriert. Dabei besteht eine unmittelbare, unverzögerte Wissensspirale/-übernahme, wobei die bearbeitete Technik auf sich selbst angewendet wird (s. Kap. 2.2.).<sup>88</sup>

Bei der Bearbeitung der IuK-technischen Infrastruktur, wird die letzte stabile Version, die über das Internet erreichbar ist, währenddessen verwendet. Werden z.B. Anwendungsprogramme wie Office-Programme u.a. weiterentwickelt, wird nicht auf diese zugleich zurückgegriffen, d.h. diese Programme werden zwar genutzt, aber nicht während ihrer Bearbeitung. Bei der IuK-technischen Infrastruktur, Anwendungsprogrammen etc. sind die Softwareentwickler auch Nutzer der Software, weswegen hier eine eher mittelbare, leicht verzögerte Wissensspirale vorliegt.

Folglich werden außer bei der Weiterentwicklung der grafischen Benutzeroberfläche und Unterstützungssoftware alle unterstützenden Werkzeuge als stabile Versionen verwendet. Damit wird die Komplexität reduziert, da sonst Fehlermeldungen auf verschiedene fehleranfällige Softwareprogramme zurückgeführt werden müssten, was mit viel Arbeitszeit verbunden ist. Die unverzögerte Wissensspirale bei der grafischen Benutzeroberfläche und Unterstützungssoftware führt zudem zu Effizienzgewinnen. Für andere inhaltliche Arbeiten wie der Nutzung von Office-Programmen z.B. zum Schreiben oder Berechnen werden stabile Versionen sämtlicher unterstützender Software (Betriebssystem, grafische Benutzeroberfläche, Office-Programme) für eine effiziente Bearbeitung bevorzugt.

Manchmal greifen KDE-Mitarbeiter auf Software z.B. Editoren, Compiler als Programmierunterstützungssoftware aus anderen Open Source-Projekten zurück. Sie integrieren u.U. auch Quellcode aus anderen Open Source-Projekten in ihren Quellcode, wenn dieser auch unter der GPL steht. Dabei entsteht eine stark verzögerte und umständliche Wissensspirale, da die KDE-Mitarbeiter in dem Fall selbst Nutzer sind. Die Übernahme des Quellcodes oder die Nutzung einer Software ist mit der Aneignung von weiterem Wissen verbunden, da den KDE-Mitarbeitern die Software aus einem anderen Projekt als Black Box erscheint. Die Wissensaneignung wird aber durch Dokumentationen, ähnliche Programmiersprache und gleiche GPL-Lizenz gegenüber der Aneignung zwischen Unternehmen stark erleichtert.

Die oben beschriebene gleichzeitige Nutzung und Bearbeitung von Software funktioniert nur, wenn Nutzer zugleich Produzenten sind (vgl. Hippel 2001). Damit beeinflusst und unterstützt die IuK-Technik z.B. als IuK-technische Infrastruktur nicht nur die Softwareent-

---

<sup>87</sup> Wenn die IuK-technische Infrastruktur überarbeitet wird, wird die neue Version auf Basis der alten Version erstellt, wobei die alte Version genutzt wird.

<sup>88</sup> Dies ist mit ein Grund, warum die Open Source-Projekte sich in einer solchen Geschwindigkeit entwickeln und weiterentwickeln konnten, so dass sie als eine Konkurrenz zu proprietären Softwareherstellern wie Microsoft gesehen werden.

wicklung, sondern das Produkt und die luK-technische Infrastruktur werden nach den eigenen Bedürfnissen geformt und angepasst. Diese Verbindung von Nutzung und gleichzeitiger Bearbeitung ist kein Novum, sondern trat in der Geschichte schon öfter auf, z.B. bei der Entwicklung von Surfing Equipment (vgl. Meyer 2003; Shah 2000; Franke, Shah 2003). Neu aber ist, dass dies unmittelbar möglich ist und sich auch eigentlich unbeteiligte individuelle Endnutzer bzw. Konsumenten an der Arbeit über transparente luK-Technik durch Einsenden von Fehlerberichten über das Fehlerberichtssystem oder von Quellcode-schnipseln zur Fehlerberichtigung beteiligen können. Dabei machen einmalige Einsendungen von Endnutzern einen Großteil aller Einsendungen aus (vgl. Mockus et al. 2002; Jörgensen 2001). Damit treiben u.a. individuelle Endnutzer die Wissensspirale mit Wissensweitergabe und Arbeitseinsatz durch (zusätzliche) Einsendungen darüber hinaus an, wobei die Anzahl der Einsendungen von dem Interesse der Endnutzer an dieser Software bzw. der Sichtbarkeit für die Endnutzer abhängt. Die grafische Benutzeroberfläche und manche Anwendungssoftware wie das Office Paket werden unterstützt, die luK-technische Infrastruktur und die Unterstützungssoftware kaum bzw. gar nicht.

Die genutzte objektorientierte Programmiersprache C++ wiederum erleichtert über einen modularen Aufbau bei der grundlegenden Qt-Bibliothek und die Endprodukte des KDE-Projekts eine Wissensweitergabe. Bisher programmierte Module ermöglichen eine direkte Übernahme in neuen Quellcode, wobei eine zeitaufwändige Überarbeitung durch standardisierte Schnittstellen nicht nötig ist. Außerdem besteht durch die textbasierte luK-technische Infrastruktur eine fast medienbruchlose Kommunikation und Produktion, wie in anderen virtuellen Gruppen (vgl. Runkehl 1998), die die Wissensweitergabe stark erleichtert.

Nur ein bestimmter Anteil der Software wird im Rahmen des KDE-Projekts ständig und unmittelbar verändert, die luK-technische Infrastruktur bleibt als unterstützende Technik für die Softwareentwicklung als Arbeitsprozess und für eine dabei ablaufende Wissensübertragung/-spirale unverändert und stabil. In den folgenden Unterkapiteln wird die unterstützende luK-technische Infrastruktur im KDE-Projekt fokussiert und vorgestellt, die größtenteils auf selbsterstellter Software basiert. Die Vorstellung der luK-technischen Infrastruktur basiert besonders auf der Website-/Homepageauswertung<sup>89</sup> und den Interviews. Mit der Vorstellung wird der Zusammenhang zwischen luK-technischer Infrastruktur und der Softwareentwicklung verständlicher, der später im Kapitel 5.4. mit dem Fokus auf der Softwareentwicklung als Arbeitsprozess untersucht wird. Dabei determiniert die unterstützende luK-technische Infrastruktur die Handlungen während der Softwareentwicklung nicht, sondern ermöglicht bestimmte Handlungsweisen.

Die verwendete Technik kann nach ihrer hauptsächlichen Kommunikations-, Informations- und Produktionsausrichtung eingeteilt werden. Dabei sind die Ausrichtungen der luK-technischen Infrastruktur wie die Tätigkeits- und Strukturbereiche des KDE-Projekts (s. Kap. 5.2.3.) nach dem Zentrum-Peripherie-Prinzip strukturiert. Die Mailinglisten und ein Chatsystem (Inter Relay Chat (IRC)) bestimmen die Kommunikationsausrichtung (s. Kap. 5.3.1.). Die Website des Projekts ist mit der Informationsausrichtung verbunden (s. Kap. 5.3.2.).<sup>90</sup> Die Produktionsausrichtung basiert vor allem auf einem Versionsmanagement-/Dateiablagensystem (s. Kap. 5.3.3.), die durch ein Fehlermeldesystem und einige weitere

---

<sup>89</sup> Im Folgenden wird unter Homepage nicht nur eine Seite, sondern ggf. alle Seiten einer Website verstanden.

<sup>90</sup> Die Kommunikations- und Informationsfunktionen sind analytische Aufteilungen, da die Informationsfunktion auch bei den Mailinglisten durch deren Archive und die Kommunikationsfunktion auf der Homepage durch einen journalistischen Teil auftritt.

entwicklungsunterstützende Tools angereichert wurden (vgl. Becher et al. 2005; für eine Einschätzung aus dem KDE-Projekt: Ettrich 2004). Diese IuK-technische Infrastruktur wird im Allgemein auch von anderen Open Source-Projekten genutzt (vgl. Yamauchi et al. 2000; Jörgensen 2001; Feller, Fitzgerald 2000, 2002). Im KDE-Projekt sind aber keine Werkzeuge vorzufinden, die einen Wahlvorgang nach dem Mehrheitsprinzip wie im Gnome-Projekt, Vergleichs-/Matchinginstrumente für Anbieter und Nachfrager wie bei Online-Auktionen/Marktplätzen (ebay) oder eine Zuweisung einer Person zu einer Arbeitsaufgabe/Arbeitsstelle wie bei Online-Jobbörsen automatisiert unterstützen (vgl. Kollmann 2001: 27; Becher et al. 2005).

Die Kommunikations-, Informations- und Produktionsausrichtung wird außerdem nach den Merkmalen der Technik als Medium (s. Kap. 2.2.3.) untersucht. Dabei ist in allen drei Ausrichtungen eine one-to-many- und many-to-one-Kommunikation gleichermaßen vorhanden, wobei aber bei der Informationsausrichtung die one-to-many-Kommunikation dominiert (vgl. Kollmann 2001: 27; Runkehl 1998). Eine identifizierbare Adresse ist in allen drei technischen Ausrichtungen vorhanden und ermöglicht auch eine point-to-point- bzw. one-to-one-Kommunikation sowie Zugriffsrechte zu spezifischen Werkzeugen. Umgekehrt erzwingen die spezifischen Zugriffsrechte eine wiederholte Nutzung einer identifizierbaren Adresse. Zusätzlich wird eine räumlich und zeitlich synchrone und asynchrone Kommunikation und Produktion ermöglicht, d.h. eine räumliche und zeitliche Entkopplung.

### **5.3.1. Kommunikationsinfrastruktur**

Die asynchrone räumliche und zeitliche Kommunikation basiert auf textbasierten Mailinglisten, dem wichtigsten Kommunikationsorgan im Projekt mit den meisten Diskussionen und Entscheidungen für die Softwareproduktion.<sup>91</sup> Die nach Themen geordneten Mailinglisten unterteilen sich in wichtige, zentrale Mailinglisten mit allgemeiner fachlicher Diskussion im Gesamtprojekt und spezielle periphere Mailinglisten für die spezifische Diskussion in den verschiedenen Subprojekten, übergreifenden Tätigkeitsbereichen bzw. für Themen mit spezifischen Aufgaben. So existieren für Tätigkeitsbereiche wie Edutainment, Übersetzung etc. je eine eigene Mailingliste für die Abstimmung untereinander. Auch die Mailinglisten können im Zusammenhang mit ihren Subprojekten nach Wichtigkeit für das Gesamtprojekt in ein Zentrum oder in eine Peripherie eingeordnet werden. Es gibt zentrale Mailinglisten respektive Subprojekte bzgl. des Desktops mit vielen Beteiligten und einer hohen Anzahl an Quellcodezeilen sowie periphere Mailinglisten bei Subprojekten für spezifische Anwendungen. Selbst die Kommunikation in den Mailinglisten ist durch ein Zentrum und eine Peripherie bestimmt, da nur wenige der Angemeldeten dort auftreten und eine große Anzahl nicht oder selten kommuniziert. Außerdem werden nur selten direkte Emails zwischen KDE-Projektmitgliedern ausgetauscht, fast alle Emails laufen über die Mailingliste, was auch bei anderen Open Source-Projekten wie im FreeBSD-Projekt und dem GNU GCC-Projekt festgestellt wurde (vgl. Yamauchi et al. 2000). Außerdem dürfen nur reale und authentische, d.h. identifizierbare, Personen an der Kommunikation teilnehmen, was generell auch für andere virtuelle Gruppen gilt (vgl. Döring 2003).

Das textbasierte Chatsystem erlaubt die räumlich asynchrone, zeitlich synchrone textbasierte Kommunikation, da dort die Anwesenden zeitgleich im Chatsystem und dadurch identifizierbar sind. Außerdem ist das Medium flüchtig, da Geschriebenes sofort gelöscht

---

<sup>91</sup> Die zeitlich eigentlich asynchrone Mailinglisten-Kommunikation ähnelt manchmal einer zeitlich synchronen, da Diskussionen bzw. Konflikte von geringer zeitlicher Dauer sind, d.h. wenige Minuten bzw. Stunden bis zu über zwei Tage ablaufen.

wird.<sup>92</sup> Der Kreis der Empfänger kann durch technische Möglichkeiten klein gehalten werden. Das Chatten wird eher von wenigen Personen und eher für private Kommunikation verwendet. Einige Entwickler nutzen während ihrer Arbeit an der Software auch das Chat-system und tauschen sich während der Arbeit zu Hause oder im Büro über die Softwareentwicklung oder private Dinge aus, die z.B. während der Arbeit passieren. Videokonferenzen werden nicht genutzt, da nicht jedes KDE-Mitglied die nötige Hard- und Software und Bandbreite hat, um dieses Werkzeug zu verwenden.

### **5.3.2. Informationsinfrastruktur**

Die Informationsausrichtung wird durch die textbasierte Website/Homepage repräsentiert, wobei diese Informationen in der Form einer one-to-many-Kommunikation für Projektexterne, aber auch für Projektbeteiligte bereit hält. Interessierte können einerseits wichtige Anleitungen (z.B. diverse FAQs<sup>93</sup>), Einweisungen, Regelungen<sup>94</sup>, etc. zur Programmierung, Dokumentation oder Übersetzung, andererseits auch wichtige Terminplanungen (z.B. den Releaseplan), Arbeitslisten, Zuständigkeiten für bestimmte Arbeitseinheiten oder Softwareteile einsehen. Neben diesem eher statischen und periphereren Bereich gibt es ein zentrales Informationsorgan, eine moderierte News-Seite, für wichtige selbst erstellte, aber auch fremde Neuigkeiten über die Entwicklung des Projekts mit einer Kommentierungsmöglichkeit. Eigene Foren unterstützen eine zusätzliche Kommunikation zwischen den Projektmitarbeitern untereinander sowie mit Projektexternen. Die Website/Homepage fungiert zudem als Zugang zu „Wissensspeichern“, da man darüber auf die Mailinglisten-Archive mit früheren Diskussionen und Entscheidungen, aber auch auf das Versionsmanagementsystem lesend zugreifen kann. Zudem wird die Website/Homepage über das Versionsmanagementsystem verwaltet und weiterentwickelt.

### **5.3.3. Produktionsinfrastruktur**

Ein textbasiertes Dateiablage- bzw. Versionsmanagementsystem zur Verwaltung des Quellcodes, aber auch von Übersetzungen, Bildern/Grafiken und den Webseiten der Website, dominiert die Produktionsausrichtung und ist das Rückgrat der Softwareentwicklung. Das Versionsmanagementsystem erlaubt durch den Zugriff über ein spezielles Werkzeug ein räumlich und zeitlich synchrones und asynchrones Arbeiten einer sehr großen Anzahl an Projektbeteiligten an demselben Quellcode, wobei auf dem eigenen Rechner vor Ort programmiert und dann in die Dateiablage zurückgespielt wird. In diesem System ist der gesamte Quellcode archiviert, so dass problematischer Quellcode gelöscht und alte Zustände wiederherstellbar sind. Schreibzugriff darauf haben nur die KDE-Projektmitglieder; Lesezugriff ist auch für Projektexterne möglich.

Das Versionsmanagementsystem vor allem für den Quellcode ist modular aufgebaut, so dass einzelne zusammenhängende Quellcodepakete an einem gemeinsamen Ort, d.h. in 79 Modulen, im System lagern. Eine Auswertung durch Robles (2004) zeigt, dass seit der Gründung von KDE im Median 11,6 EntwicklerInnen an einem Modul gearbeitet haben. Das Zentrum des Quellcodes stellt eine Basisbibliothek Qt, die von der Firma Trolltech zur Verfügung gestellt wird, und weitere projektinterne Hauptbibliotheken dar. Nebenbibliothe-

---

<sup>92</sup> Manche KDE-Mitglieder protokollieren ihre Chats. Der Zugriff ist aber auf den Protokollierenden beschränkt gegenüber Mailinglisten, die archiviert werden.

<sup>93</sup> FAQ heißt eigentlich Frequently Asked Questions und ist eine Art Hilfestellung, die Antworten für häufig gestellte Fragen bereitstellt.

<sup>94</sup> Z.B. Styleguides zur Programmierung, zum Aussehen der Programme, zu Grafiken (Icons), etc.

ken und Programme, die auf den Hauptbibliotheken aufbauen, zählen zur Peripherie.<sup>95</sup> Dabei sind Bibliotheken und Programme von zentral bis peripher gestaffelt (s.o.).

Neben dem Versionsmanagementsystem existiert ein Fehlerberichtssystem, das eine Standardisierung der Fehlerberichte ermöglicht und die Programmierer bei der Fehlersuche unterstützt. Projektmitglieder und projektexterne Produktnutzer übermitteln darüber Fehlerberichte an die verantwortlichen Projektleiter oder die Mailingliste des Subprojekts. Jeder kann dabei die Fehlerliste für jedes Versionsmanagementmodul auf der Website/Homepage abrufen. Andere produktionsunterstützende Werkzeuge, die z.T. auch im KDE-Projekt entwickelt werden (s.o.), sind z.B. ein Debugger, Compiler, spezielle Editoren für die Programmierung/Dokumentation/Übersetzung, eine grafische Versionsmanagement-Navigationsoberfläche sowie ein automatisches Übersetzungssystem.

#### **5.4. Koordinationsformen im Open Source-Projekt KDE**

In diesem Unterkapitel stehen die Koordinationsformen im Open Source-Projekt KDE im Sinne der Fragestellung im Mittelpunkt. Dabei geht es vor allem um die empirischen Untersuchungsergebnisse zu Ausprägung und Stellenwert der Koordinationsformen im KDE-Projekt, die sich in einer spezifischen Kombination darstellen. Die Begriffe Stellenwert oder Bedeutung der Koordinationsformen beziehen sich in diesem Zusammenhang auf die relative Wichtigkeit für eine Befolgung der Koordinationsformen als Institutionen, d.h. auf die Stärke ihrer Institutionalisierung. In einem idealisierten Arbeitsprozess der Softwareentwicklung mit Teilprozessen (s. Kap. 2) werden Koordinationsformen untersucht.

Bisher wurde nur in Teilbereichen der Open Source-bezogenen Literatur auf die Institutionen bzw. Normen in den Open Source-Projekten eingegangen und nur einzelne Koordinationsformen untersucht, wie im Folgenden in Kapitel 5.4.1. der Stand der Forschung für die Koordinationsformen in der Open Source-Literatur aufzeigt. Eine detaillierte und systematische Untersuchung erfolgte nicht. Zudem wurde die Beziehung zwischen IuK-Technik und Organisationsformen in organisationstheoretischer, aber nicht in Open Source-bezogener Literatur<sup>96</sup> thematisiert. Es fehlte ein Bezug auf Koordinationsformen. Daher wird in Kapitel 5.4.2. der Stand der Forschung zur Nebenfragestellung nach dem Einfluss von IuK-Technik auf das KDE-Projekt als Zusammenhang zwischen IuK-technischer Infrastruktur und Koordinationsformen untersucht.

In Kapitel 5.4.3. werden dann die empirischen Erkenntnisse zu Ausprägung und Stellenwert der Koordinationsformen in den Teilprozessen der Softwareentwicklung im KDE-Projekt vorgestellt, wobei auf die Softwareentwickler fokussiert wird. Die Teilprozesse bestehen dabei aus „Eintritt/Rekrutierung“, „Arbeitsplanung (Zieldefinition, Arbeitsverteilung von Arbeitspaketen mit detaillierter Entwicklungsplanung, Arbeitsverteilung von Posten)“, „Leistungserstellung mit detaillierter Entwicklungsplanung“ und „Arbeitszusammenführung“. Dabei wird der Zusammenhang zwischen IuK-technischer Infrastruktur und Koordinationsformen implizit untersucht.

##### **5.4.1. Koordination in Open Source-Projekten – Stand der Forschung**

In diesem Unterkapitel wird die Literatur über Open Source-Projekte nach Anzeichen für Koordinationsformen durchsucht, die in bestimmten Situationen Anwendung finden. Die

---

<sup>95</sup> Das untersuchte Open Source-Projekt KDE hat seine Anwendungssoftware unter der GPL lizenziert, die Bibliothekssoftware aber unter einer weniger restriktiven Lizenz, die die Nutzung von Softwareteilen in proprietärer Software erlaubt. Die Basisbibliothek der Firma wiederum ist doppellizenziiert worden (vgl. Välimäki 2003), d.h. sie kann unter der GPL und der proprietären Lizenz verwendet werden.

<sup>96</sup> (ausgenommen Tuomi 2001)

Literatur über Open Source-Projekte ist sehr heterogen, wobei zu diesem Aspekt wenig Literatur existiert, obwohl in den letzten Jahren vor allem im englischsprachigen Bereich viel zu Open Source-Projekten publiziert wurde (s. [http://opensource.mit.edu/online\\_papers.php](http://opensource.mit.edu/online_papers.php)). Dabei kann die Literatur in unterschiedliche, sich z.T. überschneidende Bereiche eingeteilt werden.

Autoren vor allem aus der Informatik-Disziplin konzentrieren sich zum einen auf den Softwareentwicklungsprozess als Phasenmodell (s. Kap. 2.3.2.3.) und die Analyse der produzierten Software, wobei die Programmsicherheit und die Wechselwirkung mit anderer Software untersucht wird (vgl. Jörgensen 2001; Mockus et al. 2002; Feller, Fitzgerald 2002: 83ff.; Reis, Fortes 2002; Kautz, Ramzan 2001; Scacchi 2002a, 2002b). Zum anderen untersuchen sie Verhaltensspuren, die in den Versionsmanagementsystemen vorliegen (vgl. González-Barahona, Robles-Martínez 2003; Ghosh, David 2003; Ghosh, Prakash 2000; Koch, Schneider 2002). Weiter gehen einige Autoren aus der Rechtswissenschaft auf die rechtlichen Grundlagen des Open Source-Ansatzes ein und versuchen einen Bezug zwischen Lizenzen und Copyright-Gesetzen herzustellen, die Relevanz dieser Vertragsart herauszustellen und Folgerungen bezüglich neuer Copyright-Gesetze abzuleiten (vgl. z.B. Jaeger, Metzger 2002; Benkler 2000; Söderberg 2002). Bei eher betriebswirtschaftlich orientierten Autoren steht die Beziehung zwischen Open Source-Projekten und Unternehmen im Mittelpunkt, d.h. die effiziente, wirtschaftliche Verwendungsmöglichkeit und die Businessmodelle von Open Source-nahen Unternehmen (vgl. z.B. Busa et al. 1999; Wegberg, Berends 2000; Pal, Madanmohan 2002).

In einer Mikroperspektive steht die Motivation der Softwareentwickler in Open Source-Projekten im Mittelpunkt, wie Spass, Lernen, Arbeitsmarktinvestitionen und Engagement für Open Source (s. Anfang Kap. 5.2.2.; für eine Zusammenfassung der Motivationen s. Rossi 2004: 3ff.; Osterloh 2002; Weber, S. 2004: 128ff.; Holtgrewe, Brand 2007). In einer übergreifenden Perspektive auf Open Source-Projekte werden überwiegend von Autoren der politischen Ökonomie und der Informatik formale und informale Normen und Werte wie Hackerethik, technische Rationalität, Führungsstruktur und Reputation festgestellt (vgl. Weber, S. 2004: 157ff.; Holtgrewe, Brand 2007). In dieser Arbeit interessiert aber der Arbeitsprozess zwischen der individuellen Ebene und der Projektebene.

Dabei werden aus dieser übergreifenden Perspektive auf Open Source-Projekte *Macht- und hierarchische Elemente* erkannt (vgl. Rossi 2004: 15f.). Auf der einen Seite beschreiben empirische Fallstudien über Open Source-Projekte eine Führungsstruktur von Projektleitern, auf der anderen Seite eine Kerngruppe von Softwareentwicklern mit hoher Reputation (s. z.B. Bergquist, Ljungberg 2001; Mockus et al. 2002; Bezroukov 1999a, b; German 2003). Die sogenannten Projektleiter haben verschiedene Funktionen inne, wie eine Vorbildfunktion, Verbreitung einer Vision, Zusammenhalten des Projekts und Projektbeteiligte an wichtige Probleme heranzuführen (Lerner, Tirole 2000: 21). Ein Release, eine stark strukturierte Phase der Entwicklung, wird durch einen Releasekoordinator koordiniert (vgl. Jörgensen 2001).

“When the project nears a product release, one of the core developers volunteers to be the release manager, responsible for identifying the critical problems (if any) that prevent the release, determining when those problems have been repaired and the software has reached a stable point, and controlling access to the repository so that developers don't inadvertently change things that should not be changed just prior to the release. The release manager creates a forcing effect in which many of the outstanding problem reports are identified and closed, changes suggested from outside the core developers are applied, and most loose ends are tied up. [...] The role of release manager is rotated among the core developers with the most experience with the project” (Mockus et al., 2002: 322).

Außerdem besteht z.B. bei Linux eine wohlwollende Diktatur („Benevolent Dictatorship“), bei der ein Gründer im Allgemeinen entscheidet und Teile seiner Entscheidungsgewalt an vertrauenswürdige Untergebene („Trusted Lieutenants“) abgibt (vgl. Raymond 1998b; Ljungberg 2000).

Ein Statusunterschied durch Reputation zeigt sich in Open Source-Projekten durch kleine Gruppierungen, die an einem Großteil der Kommunikation und Produktion beteiligt sind oder diese initiiert haben (vgl. Moon, Sproul 2000; Mockus et al. 2002; Koch, Schneider 2002; Dempsey et al. 1999; Ghosh, David 2003; für eine Aufarbeitung s. Rossi 2004: 14ff.).<sup>97</sup> Insgesamt wird nur bedingt auf die verschiedenen hierarchischen Strukturen mit ihrem Auftreten, ihrem Stellenwert oder ihrer konkreten Anwendung im Arbeitsprozess eingegangen. Eine Begrenzung der Hierarchie bzw. Macht zeigt aber Raymond (1998a) auf, der über die Abwesenheit einer zentralen Entscheidungsfindung bei den Projekten berichtet. Diese verschiedenen Aspekte werden in der neuen Definition in Kapitel 3.2. beachtet.

*Kooperation* wird auch thematisiert, aber eher im Gewande eines Tauschmarkts von Quellcode bzw. Ideen gegen Reputation (z.B. „Cooking Pot“ Market) von Raymond (1998a, 1999a), Ghosh (1998), Weber (2000), Lerner/Tirole (2000, S. 14ff.) und Franck/Jungwirth (2002a, b) beschrieben. Eine andere Sichtweise knüpft an Mauss (1984) oder an Blau (1964) an und sieht einen reziproken Austausch in Form von Gabe und Gegengabe von Quellcode als Kooperation, wobei der „Beschenkte“ auch das Open Source-Projekt darstellen kann. Dabei werden z.B. Programmierfehler von anderen Mitgliedern korrigiert und Konsens spielt eine Rolle (vgl. Zeitlyn 2003; Bergquist/Ljungberg 2001; Iannacci 2003; Working Group 2000: 15f.).<sup>98</sup> Entgegen der Meinung vieler Autoren spielt Kooperation in Open Source-Projekten eine besonders wichtige Rolle, da in Artikeln über virtuelle Gruppen, zu denen auch die Open Source-Projekte gehören, Kooperation als bedeutend angesehen wird (vgl. Kollock 1999; Levine 2001). Daher wurde in Kapitel 3.2. eine neue Definition angestrebt, die unabhängig von unpassenden Vorannahmen wie der Marktmetapher sind.

Ein *Wettbewerb* um (zukünftige) Arbeitsplätze wird vor allem im Rahmen der Motivation debattiert, hat aber keinen so hohen Stellenwert bei den Open Source-Mitarbeitern wie ihn die Literatur suggeriert.<sup>99</sup> Ein Wettbewerb um Reputation im Projekt wurde zwar von Raymond (1998b) andiskutiert, aber in der Literatur nicht dezidiert untersucht, eher implizit angenommen.

---

<sup>97</sup> Diese Unterscheidung in Zentrum und Peripherie als „town council effect“ wurde schon von Alan Cox im Hinblick auf Linux gesehen, wobei sich eine Hierarchie zwischen den Leistungserstellern und dem nur diskutierenden Rest herausbildete (nach Bezroukov 1999a).

<sup>98</sup> „Developer A posts a bug report on the kernel mailing list in order to get the bug fixed. Developer B provides the bug fix and is, in turn, reciprocated by developer C when he posts his own bug report“ (Iannacci, 2003).

<sup>99</sup> Online-Befragungen bestätigten die geringe Bedeutung der Arbeitsmarktinvestitionen bei der Frage nach der Motivation für eine Beteiligung (vgl. Lakhani, Wolf 2003; Robles et al. 2001). Schon Raymond (1998b) schrieb: „Occasionally, the reputation one gains in the hacker culture can spill over into the real world in economically significant ways. It can get you a better job offer, or a consulting contract, or a book deal. This kind of side effect, however, is at best rare and marginal for most hackers; far too much so to make it convincing as a sole explanation [...]“. Außerdem konnte festgestellt werden, dass sich Softwareentwickler unterschiedlich engagieren, weswegen Reputation und damit die Chancen auf einen Arbeitsplatz unterschiedlich ausgeprägt sind. Es kann also von einem allgemeinen Signal auf dem Arbeitsmarkt nicht gesprochen werden (vgl. Demil, Lecocq 2003: 14). Außerdem wurde in Interviews erklärt, dass die projektinterne Reputation nur für Unternehmen mit einer thematischen und personellen Nähe zum Open Source-Projekt verständlich ist.

In diesem kurzen Überblick über die Literatur zu Open Source-Projekten zeigt sich, dass bestimmte Formen der Koordination in diesen Projekten eine Rolle spielen, aber nicht konkret von den Autoren thematisiert werden. So wird vor allem von einer hierarchischen Koordination gesprochen (vgl. Rossi 2004: 15ff.), die Kooperation vor allem durch die Brille eines rationalen Akteurs auf einem Markt betrachtet. Wettbewerb tritt zwar verdeckt auf, wird aber nicht dezidiert untersucht. Eine gemeinsame, übergreifende, vergleichende Betrachtung bzw. Gegenüberstellung der drei Koordinationsformen erfolgt nicht. Diese Defizite werden aufgegriffen und im Weiteren bearbeitet. Es fehlen dabei nicht nur eine Definition der Koordination an sich, sondern auch konkrete Definitionen der Koordinationsformen, womit sich aber schon in Kapitel 3.2. befasst wurde.

Eine Zwischenperspektive hinsichtlich der übergreifenden Projekt-Perspektive und einer individuellen Perspektive, die situative Praktiken mit ihren Normen sowie soziale Kollaborationsbeziehungen und Sanktionen untersucht, wurde bisher nicht eingenommen (vgl. Holtgrewe, Brand 2007). Neben dem Bezug auf Normen in situativen Praktiken bei Holtgrewe und Brand (2007), stellt die Untersuchung des Softwareentwicklungsprozesses in den Projekten durch Autoren aus der Informatik eine Annäherung an situative Praktiken dar (s.o.). Bei diesen Autoren werden jedoch Normen nicht dezidiert untersucht (s. z.B. Jörgensen 2001; Mockus 2002; German 2003). Folglich wurden keine Koordinationsformen in dieser Zwischen- und Handlungsperspektive bzw. in den Teilprozessen der Softwareentwicklung betrachtet, was in dieser Arbeit in Kapitel 5.4.3. untersucht wird.

#### **5.4.2. Zusammenhang zwischen IuK-technischer Infrastruktur und Koordinationsformen - Stand der Forschung**

In diesem Unterkapitel wird der Stand der Forschung zur Teilfragestellung nach dem Zusammenhang zwischen der IuK-technischen Infrastruktur im Open Source-Projekt KDE und den Koordinationsformen untersucht, die sich aus der Frage nach dem Einfluss der IuK-Technik auf die Softwareentwicklung ergab. Bisher besteht keine Literatur zum Zusammenhang zwischen IuK-technischer Infrastruktur und Koordinationsformen in Arbeitsnetzen oder Open Source-Projekten, weswegen zuerst auf die Organisationsforschung abgestellt wird, die sich aber eher auf Unternehmen konzentriert. Allgemeine Tendenzen der Nutzung von IuK-Technik können aber auch herangezogen werden.

Aus der technikphilosophisch-techniksoziologischen Literatur in Kapitel 2.2. ist bekannt, dass Technik nicht soziales Handeln determiniert, aber bestimmte Verwendungsweisen mit Zweck-Mittel-Beziehungen intendiert. Nach Orlikowski (1992) wurde aber gerade in der Organisationsforschung ein deterministischer Blick hinsichtlich des Einflusses von Technik auf Organisationsstrukturen eingenommen. Beim Neo-Institutionalismus wurde nur ein schwacher oder gar kein Zusammenhang zwischen der verwendeten Technologie und der formalen Organisation gefunden (vgl. Walgenbach 1999: 326ff.). Die neue Institutionenökonomie kommt zu ähnlichen Ergebnissen. Hier konnte bisher nicht festgestellt werden, ob die IuK-Technik eher zu einem Move-to-the-Market, -Middle oder -Hierarchy führt, d.h. eine bevorzugte Organisationsform besonders unterstützt (vgl. Malone et al. 1987; Brynjolfsson et al. 1988; Clemons et al. 1993). Auch bei der empirischen Untersuchung von Koordinationsformen auf dem elektronischen Arbeitsmarkt, die von der neuen Institutionenökonomie angeregt wurde, konnte kein Einfluss der IuK-Technik festgestellt werden (vgl. Becher et al. 2007). Daraus lässt sich ableiten, dass die IuK-technische Infrastruktur einer Organisationsform z.B. eines Open Source-Projekts keinen determinierenden Einfluss auf das Auftreten der Koordinationsformen hat, aber u.U. bestimmte Koordinationsformen unterstützen kann.

Bei der Nutzung des Internets konnten einzelne Koordinationsformen von verschiedenen Autoren nachgewiesen werden, wobei keine Koordinationsform bevorzugt wird. Im Internet besteht neben aggressivem und diskriminierendem auch kooperatives und prosoziales Verhalten zwischen Einzelpersonen (vgl. Döring 2003: 279ff., 546ff.; Hoffmann 2000). In virtuellen Gruppen sind vor allem kooperative und hierarchische Elemente anzutreffen, Wettbewerb kann aber auch vorkommen (vgl. Kollock 1999; Levine 2001; Götzenburger, Löger 2000; Döring 2003: 490ff.).

Der Zusammenhang zwischen der IuK-technischen Infrastruktur und den Koordinationsformen wurde in der Open Source-bezogenen Literatur bisher nicht thematisiert und wird daher im Folgenden am Beispiel des KDE-Projekts untersucht.<sup>100</sup>

### **5.4.3. Koordinationsformen in den Teilprozessen der Softwareentwicklung**

In diesem Unterkapitel stehen die empirischen Erkenntnisse zu Ausprägung und Stellenwert der Koordinationsformen in den Teilprozessen der Softwareentwicklung im KDE-Projekt im Mittelpunkt. Die empirischen Erkenntnisse setzen sich aus Einzelerkenntnissen aus Interviews, Mailinglistenbeobachtung, Selbstaussagen auf der Website/Homepage und Gesprächen/teilnehmender Beobachtung auf Konferenzen zusammen. Die Ausprägungen der Koordinationsformen im Open Source-Projekt KDE werden dann anhand der Klassifikationen aus dem Kapitel 3 identifiziert. Die Bedeutung der Koordinationsformen wird anhand der Indikatoren der Institutionalisierungsstärke aus Kapitel 4 festgestellt. Zusätzlich können, da das KDE-Projekt hoch virtuell ist, die empirischen Erkenntnisse herangezogen werden, um den Zusammenhang zwischen IuK-technischer Infrastruktur und Koordinationsformen festzustellen.

Da die Softwareentwicklung im Zentrum steht und die Softwareentwickler den größten Mitgliederanteil stellen (s. Kap. 5.2.), wird vor allem auf die Softwareentwickler fokussiert. Da aber bei der Betrachtung der Technik in Kapitel 2.2. die Endnutzer einen Einfluss auf die Entwicklung des Endprodukts des KDE-Projekts haben, wird die Beziehung zwischen Softwareentwicklern und projektexternen Endnutzern mit untersucht.

Im Folgenden werden die Ausprägung und Bedeutung der Koordinationsformen in den untersuchten Teilprozessen „Eintritt/Rekrutierung“, „Arbeitsplanung (Zieldefinition, Arbeitsverteilung von Arbeitspaketen mit detaillierter Entwicklungsplanung und Arbeitsverteilung von Posten)“, „Leistungserstellung“ und „Arbeitszusammenführung“ vorgestellt. Die parallele und relativ unstrukturierte Arbeit im KDE-Projekt lässt Teilprozesse z.T. parallel zueinander ablaufen und führt dazu, dass die Teilprozesse oftmals nicht klar voneinander abtrennbar sind. Bei der Auswertung des empirischen Materials wurde klar, dass bestimmte Teilprozesse zusammengehören bzw. stark miteinander verbunden sind. Im KDE-Projekt ist die Arbeitsverteilung von Arbeitspaketen und die Leistungserstellung nicht ohne die detaillierte Entwicklungsplanung denkbar, da diese wegen der zyklischen und parallelen Softwareentwicklung miteinander verquickt sind. Die KDE-Mitarbeiter wählen zwar Arbeitspakete oder Arbeitsbereiche eigenständig aus, haben aber manchmal zugleich eine grobe Vorstellung von ihrem weiteren Vorgehen und planen mit anderen Mitarbeitern das weitere Vorgehen (als detaillierte Entwicklungsplanung). Der Teilprozess Leistungserstellung ist durch einen Versuchs- und Irrtumsprozess gekennzeichnet, wobei zwar eigenständig das Vorgehen detailliert geplant wird, aber die (detaillierte Entwicklungs-)Planung

---

<sup>100</sup> Nur Tuomi (2001) kommt dieser Fragestellung nahe, wobei er die Beziehung zwischen der Softwareentwicklung eines Produkts, der genutzten Infrastruktur und den Linux-Mitgliedern aus einer Perspektive der Actor-Network Theory genauer untersuchte. Die Mitglieder unterstützen beispielsweise die Fehlersuche und -behebung durch eine Informationsdatenbanken für Fehler oder ein Fehlerberichtssystem, aber auch über Mailinglisten.

oftmals anhand der bisher produzierten Software zeit- und problemnah mit anderen Mitarbeitern zusammen erfolgt. Im Weiteren wird daher die detaillierte Entwicklungsplanung mit der Arbeitsverteilung von Arbeitspaketen und der Leistungserstellung zusammen betrachtet.

Die Teilprozesse werden nacheinander in je einem Unterkapitel untersucht, wobei am Anfang kurz der jeweilige Stellenwert der Koordinationsformen vorgestellt wird, der sich empirisch gezeigt hat. Danach werden die Koordinationsformen nach ihrer absteigenden Bedeutung aufgeführt. Da auch die Beziehung zu den Endnutzern eine Rolle spielt, wird diese in den Teilprozessen in einem eigenen Unterkapitel analysiert, in denen diese einen Einfluss hat.

#### 5.4.3.1. Eintritt/Rekrutierung

Bisher wurde die Art und Weise des Eintritts in der Literatur über Open Source-Projekte vernachlässigt und davon ausgegangen, dass es keine Auswahl von Neumitgliedern eines Projekts gibt (s. z.B. Demil, Lecocq, 2003: 10; Rossi 2004; Jörgensen 2001; Mockus et al. 2002; German 2003). Im Allgemeinen kann jeder Interessierte an dem Open Source-Projekt KDE mitarbeiten. Anhand des empirischen Datenmaterials lässt sich zeigen, dass eine Gleichberechtigung aller potenziellen Mitglieder durch den offenen Lesezugriff auf das Versionsmanagementsystem mit dem dort lagernden Quellcode besteht. Außerdem steht das Fehlerberichtssystem für jeden offen, selbst für Projektexterne (s. <http://developer.kde.org/source/anonsvn.html>). Gleiches gilt für die meisten Mailinglisten, die einen offenen Lese- und Schreibzugriff haben (s. <http://kde.org/maillinglists/>). Damit ist ein fließender Übergang vom potenziellen zum offiziellen KDE-Mitarbeiter möglich. Zusätzlich macht das Open Source-Projekt KDE eher verhaltene Werbung über seinen Internetauftritt, von Mitgliedern geschriebene Zeitschriftenartikel, Auftritte auf Linuxkonferenzen etc. (s. Becher et al. 2005: 126f.).

Dieser fließende Übergang ins KDE-Projekt beinhaltet aber einen Grenzübertritt, der koordiniert wird. Beim Eintritt dominiert die Koordinationsform Hierarchie/Macht, da ein Subprojektleiter immer die Zustimmung zu einem Eintritt ins KDE-Projekt geben muss und dabei die Eignung kontrolliert. Zusätzlich besteht ein nachgeordneter Qualitätswettbewerb um Quellcode zwischen potenziellen und realen Mitarbeitern ohne strenge Auslese. Eine Kooperation in Form von Hilfe des Projektleiters oder andere potenzielle Mitarbeiter kann zwar auftreten, ist aber unbedeutend, da eine selbsttätige Einarbeitung erwartet wird. Die Ausprägungen und Bedeutung der Koordinationsformen wird im Weiteren untersucht.

- *Hierarchie/Macht* zeigt sich an dieser Stelle bei den (Sub-)Projektleitern. Sie sind dabei oftmals die ersten Ansprechpartner, die potenzielle Mitglieder bei vorhandener Programmierbegabung anleiten und bei minderer bis kaum vorhandener fachlicher Programmierkompetenz diese an die Übersetzer- bzw. Designprojekte weitervermitteln. In vielen Fällen werden Anfragen von potenziellen Mitarbeitern mit einem geringen Wissensstand ignoriert.

Der verantwortliche Projektleiter, so wurde in den Interviews vermittelt, überwacht immer mittels seiner Positions- und Expertenmacht den Eintrittsprozess, wobei er ihm eingesandte Quellcodebeiträge bzw. Fehlerkorrekturen von Projektexternen fast immer auf ihre Fehlerlosigkeit oder schädigenden Quellcode überprüft, da er für die Einstellung später zur Verantwortung gezogen werden kann. Ein Projektleiter kann auf die Softwareentwicklung des potenziellen Mitglieds Einfluss nehmen, indem er einerseits droht, den Quellcode nicht ins Versionsmanagementsystem einzustellen bzw. sich ggf. weigert. Andererseits kann er die Programmierung dirigieren, indem er über frühere Lösungsversuche oder Herangehensweisen (nicht) informiert. Beides wird aber selten angewendet. Selbstaussage Nr. 42 ist ein Beleg hierfür:

„That depends on what you mean with 'get involved'. After I started using KDE, there were some things I didn't quite like, so I fixed this and that and sent the patch [kleine Quellcodeänderung/Fehlerberichtigung], I also wrote the first version of [a program] at that time. I also followed developers mailing lists, but I started really seriously developing KDE [Software] only less than a year ago.“

Manche Projektleiter können durch ihr charismatisches Auftreten potenzielle Mitarbeiter von der Übernahme auch von größeren Arbeitspaketen überzeugen. Diese Fälle treten aber nicht häufig auf. Seit Gründung des KDE-Projekts werden jedem potenziellen Mitglied, nachdem durch mehrfaches Einreichen fehlerfreier Beiträge Ausdauer gezeigt wurde, als Belohnung Schreibrechte zum Versionsmanagementsystem zugeteilt (Interviews RN, DF).

- *Wettbewerb* spielt durch den Vergleich des eingesandten Quellcodes mit der Qualität des Quellcodes der angestammten Mitglieder seit der Gründung eine Rolle. Die wiederholt eingesandten Programme und Quellcodeänderungen sollen dabei ungefähr dem Standard der KDE-Mitarbeiter z.B. in der Lesbarkeit entsprechen und keine weiteren Fehler in einem vorhandenen Programm hervorrufen. Dabei gilt, dass der Eintritt auch von dem Umfang des eingesandten und funktionstüchtigen Quellcodes abhängig ist. Üblicherweise werden wenige Veränderungen von Quellcodezeilen zur Fehlerbeseitigung, seltener dagegen komplette Veränderungen von Programmen eingesandt oder funktionstüchtige, umfangreiche, neue Programme aus der KDE-Projektumgebung aufgenommen. Manchmal werden auch Schreibrechte zum Versionsmanagementsystem vergeben, wenn zwar der Quellcode fehlerbehaftet ist, aber Beharrlichkeit/Engagement zu erkennen sind.

Die geringere Bedeutung eines Qualitätswettbewerbs wird durch folgende Aussage unterstrichen: „Man muß vorher auch nicht sehr viel Code produziert haben. Es reicht, wenn jemand vernünftig ist, ein bißchen Code zum Präsentieren geschrieben hat und bisher gute Vorschläge gemacht hat“ (Interview MK). Nicht nur zum Qualitätswettbewerb, sondern auch zu Elementen der Hierarchie/Macht liefert folgende Selbstaussage Nr. 17 Informationen:

„After a while I wanted to write a file dialog for KDE, but the response from the mailinglists was that such a thing is not needed. So I searched for another project to do, and decided to start a presentation program. I posted to the KDE lists, that I'll write a powerpoint clone. As I was new to C++ and KDE, nobody took me serious. But after some time I had something which was awfull code but worked somehow. So I got a CVS account and got more and more involved into KOffice and KDE.“

Das Erringen der Aufmerksamkeit des Projektleiters gegenüber Mitgliedern mit Reputation ist für die potenziellen Mitglieder schwierig und zeigt sich hin und wieder in verspäteten Antworten des Projektleiters auf direkte Emails (s. gleiches Problem beim Debian-Projekt: Holtgrewe, Brand 2007). Potenzielle Mitarbeiter z.T. aus der Open Source-Bewegung, die von vornherein ein hohes Maß an Softwareentwicklungswissen mitbringen, arbeiten sich recht schnell ein oder haben bereits ein Programm auf Basis der KDE-Software vorzuweisen und werden schnell aufgenommen.

- Seit der Projektgründung gilt generell, dass Projektleiter bzw. andere Mitglieder bei Problemen den potenziellen Mitarbeitern mit Informationen und eigenen kleinen Aushilfsarbeiten altruistisch und mit freundlicher Kommunikation bei gelegentlichen Anfragen helfen (*Kooperation*). Dies verdeutlicht Selbstaussage Nr. 16:

„The team are very friendly and keen to help those who show they are serious about getting involved. Being new to programming, I was having difficulty with things like the now-obsolete [program], but whenever I asked for clarification on a point, I always got great feedback.“

Der Projektleiter oder Personen mit Reputation helfen vor allem in den seltenen Fällen, in denen Spezialfragen von Neulingen gestellt werden, sonst werden Fragen mit offensichtli-

chen Antworten ignoriert oder u.U. auf bestehende Lösungen hingewiesen. Hin und wieder helfen sich potenzielle Mitarbeiter und Neuaufgenommene als Gleichgestellte bei Fragen auf spezifischen Mailinglisten. Allgemein wird der schnelle Eintritt durch Verständnis für den Ablauf bzw. Werte/Normen eines Open Source-Projekts unterstützt. Es gilt die Norm der Selbstkontrolle („Netiquette“) und die Konzentration auf die (technische) Arbeit, d.h. störende Kommunikation soll unterlassen werden. Doch wird beim Eintritt eine selbsttätige Einarbeitung erwartet, was eine Kooperation einschränkt (vgl. Brand, Holtgrewe 2007).

### 5.4.3.2. Arbeitsplanung

Im Folgenden werden nacheinander die weiteren einzelnen Schritte des Teilprozesses Arbeitsplanung dargestellt: „Zieldefinition“, „detaillierte Entwicklungsplanung mit Arbeitsverteilung von Arbeitspaketen“ sowie „Arbeitsverteilung von Posten“. Wie oben beschrieben, wird die Arbeitsverteilung von Arbeitspaketen zusammen mit der detaillierten Entwicklungsplanung untersucht. Beim KDE-Projekt ist zwar der Übergang vom Teilprozess Zieldefinition der Arbeitsplanung zur detaillierten Entwicklungsplanung fließend, es wird jedoch zwischen der Planung, die das Gesamtprojekt umfasst, und der detaillierteren Planung hinsichtlich konkreter Software in den Subprojekten unterschieden. Eine Arbeitsverteilung kann dabei nicht nur in dem Teilprozess detaillierte Entwicklungsplanung, sondern auch bei der Zieldefinition auftreten. Die Projektmitglieder übernehmen aber lieber überschaubare Arbeitspakete als seltene, arbeitsaufwendige Arbeitsbereiche, die in der Zieldefinition festgelegt wurden und oft nicht allein bearbeitet werden können. Endnutzer sind bei der Zieldefinition und der Arbeitsverteilung durch Postenbesetzung nicht beteiligt, beeinflussen aber die Arbeitsverteilung von Arbeitspaketen zusammen mit der detaillierten Entwicklungsplanung.

#### Zieldefinition

Die Definition eines gemeinsamen Ziels als Teilprozess der Arbeitsplanung ist hauptsächlich durch Kooperation bestimmt, da alle KDE-Mitglieder sich an der Diskussion der Ziele für den nächsten oder übernächsten wichtigen Release beteiligen können und die Diskutanten i.A. zu einem Konsens kommen. Hierarchie/Macht hat einen mittleren Stellenwert, da einerseits die Einflussnahme von Personen mit Reputation begrenzt ist, andererseits Sanktionen eine Kooperation unterstützen. Diesen Stellenwert der Kooperation zeigen Selbstaussagen Nr. 40 und 43, wobei die Sanktionen der Hierarchie/Macht die Kooperation stützen:

„I think that KDE proves how people from all over the world can set their differences aside and volunteer and cooperate together.“ und „KDE -the team- was, is, and (hopefully) will always be a neutral zone. So far, those who have attempted to use KDE fora to express their political points of view or their hate, received a very negative response. And people off different origins, from enemy countries, with different religions coexist, creating a magnificent and robust project instead of fighting each other.“

Ein Wettbewerb der Meinungen besteht selten, wird noch seltener durch eine Mehrheitswahl mit Abstimmungsmechanismen unterstützt und hat daher kaum einen nennenswerten Stellenwert. Im Folgenden werden die Koordinationsformen genauer dargestellt.

- Eine *Kooperation* bzgl. Zieldefinition basiert auf einer diskursiven Konsenssuche, die durch den freien Schreib- und Lesezugriff auf den Mailinglisten ermöglicht wird. Ein Entzug des Schreibzugriffs ist möglich, wurde aber während des Erhebungszeitraums nicht berichtet. Der Grund liegt im Widerspruch zwischen einer technischen Ausgrenzung und der sozialen Bewegungsnorm des freien Informationsflusses.

Zieldefinitionen werden einerseits für den kommenden offiziellen Release auf breiter Basis und andererseits immer für das gesamte Projekt mit abschließendem Konsens diskutiert, was sich durch Selbstaussage Nr. 16 belegen lässt:

„It's also great to see decisions being made for practical reasons. For example, [program] was replaced by [program], which meant we lost the buzzword CORBA, but the technical reasons behind the switch are sound and have paid off in ease of development and efficiency, without us losing any important functionality.“

Es werden z.B. als Ziele eine große gemeinsame Anstrengung mit neuen Funktionen für geradzahlige, bedeutende Softwareversionen 1.0, 2.0, 3.0 (Interview RN)<sup>101</sup>, wie die Verwendung und Portierung einer neuen Programmiersprache für einen Teilbereich angestrebt. Weitere Ziele sind kleine Veränderungen für Zwischenversionen z.B. 1.1, 1.2. etc., wie eine Zusammenführung mehrerer Funktionen in eine neue Bibliothek.

Die weitere Entwicklung wird seit Gründung des Projekts, aber auf jeden Fall seit Einführung des Releasezyklus (Juli 1998) hin und wieder besprochen. Die Einführung eines Entertainmentbereichs für Anwendungsprogramme mit entsprechender IuK-technischer Infrastrukturunterstützung ist ein weiteres Beispiel für eine Entscheidung auf dieser Ebene, die unabhängig vom Release erfolgt.

Bei den Absprachen wird generell über das Vorgehen und die verschiedenen individuellen Ziele bei der Arbeit gleichberechtigt verhandelt bzw. qua besserem technischen Argument überzeugt, was als „zwangloser Zwang eines besseren Arguments“ bezeichnet werden kann (vgl. Ettrich 2004). Ziele werden ohne Ansehen der Person ausdiskutiert. Dabei wird nach Aussage der Interviewten Konsens angestrebt; es gibt keine formalen (Abstimmungs-)Regeln. Damit ist eine gruppenorientierte Kooperation bestimmend, bei der jeder KDE-Beteiligte seine Argumente vorträgt. Grundlage ist ein allgemeines Vertrauen in die Redlichkeit eines Vorschlags oder Arguments. Zusätzlich zeigt sich das gemeinsame Interesse an der Softwareentwicklung und ihrer Weiterentwicklung in der Identifikation mit der Gruppe/den Gruppenzielen, was sich z.B. in dem Logo für das Gesamtprojekt bzw. –produkt und in spezifischen Logos zentraler Software (z.B. Konqui, der Drache, für Web- und Dateibrowser Konqueror) oder in dem Stolz bei deren Präsentation auf Linux-Konferenzen widerspiegelt. Dadurch kommen auch gegenseitige Abhängigkeiten vor, die in den anderen Teilprozessen wie Arbeitsverteilung von Arbeitspaketen/detaillierte Entwicklungsplanung, Leistungserstellung und Arbeitszusammenführung auftreten.

Eine reziproke Kooperation besteht manchmal in der Form, dass eine Absprache getroffen wird, gemeinsam ein größeres Ziel zu verfolgen (Emailinterview RN). Ein Softwareentwickler mit hoher Reputation schrieb in diesem Zusammenhang:

„Jedenfalls dachte ich, sowas willst du aber zum arbeiten haben, das gibts nicht, also programmierst du eine Entwicklungsumgebung für KDE. Daraufhin in die Mailinglisten eingetragen, dahin gepostet was ich machen will und ob's da noch andere leute gibt oder ob die das für sinnvoll halten. Ich wollte ein paar tage auf kommentare warten, die dann auch kamen, und freitags gab's die Ankündigung von KDevelop 0.1 auf der Mailingliste. Das hab ich mir angesehen und (wie der Rat der anderen auch schon war - mach irgendwo mit aber mach nix alleine, das gibt sonst nix) dann [KDE-Mitglied] angeschrieben und meine Hilfe angeboten.“ (Emailinterview RN).

Bei einem größeren Konflikt über längere Zeit, d.h. mehrere Tage, wird ein Konsens durch einen freiwilligen unparteiischen Schlichter herbeigeführt. In manchen Fällen zeichnet sich eine grobe konsensuale Entscheidung ab, wobei ein Freiwilliger, oft aus dem inneren Kreis, den Spielraum nutzt und das Problem im grob abgesteckten Rahmen eher in seinem Sinne löst.

---

<sup>101</sup> s. Release Schedule: <http://en.wikipedia.org/wiki/KDE>

- Der Einfluss der Projektmitglieder mit Reputation, die u.a. auch Posteninhaber sind, ist groß, da sie oftmals die besseren technischen Argumente bzw. fachliche Kompetenz einbringen können, die aus den Erfahrungen und dem Wissen über das Projekt und die Software entstanden sind. Die Übernahme eines Postens, wie in Unternehmen oder im Linux-Projekt mit Linus Torvalds als oberstem Posteninhaber („wohlwollender Diktator“), führt im KDE-Projekt nicht automatisch zur Durchsetzung von *Hierarchie/Macht*. Die Expertenmacht bzw. Reputationsmacht hat auch Grenzen, da sich die kompetenten Reputations-träger nicht immer durchsetzen und niemanden zwingen können, ihre Meinung zu übernehmen. Eine körperliche Bestrafung, die z.B. Grundbedürfnisse wie Essen, Trinken, etc. einer Person berührt, ist wegen der hohen Virtualität in einem Open Source-Projekt nicht anzutreffen. Dabei kontrolliert ein Posten in Form eines Mailinglistenmoderators oder Subprojektleiters die Emailinhalte üblicherweise nicht und sanktioniert auch nicht durch Entzug des Lese- und Schreibzugriffs. Wahrscheinlich ist damit zu viel Arbeitszeit verbunden, die von der Softwareentwicklung ablenkt. Außerdem kann damit ein grundsätzlicher Konflikt zwischen projektschützender Bestrafung und der Norm des freien Informationsflusses/der Transparenz umgangen werden.

Projektmitglieder mit Reputation können aber hin und wieder über die Informationsmacht eingreifen, indem sie Themen auf die Diskursagenda setzen bzw. durch entsprechende Argumente verhindern oder wichtige Informationen aus sozialen Netzwerkkontakten z.B. zu anderen, externen Open Source-Projekten weiterleiten bzw. nicht weiterleiten. Die Entwickler entscheiden auch manchmal, ob die oben genannten Wunschfunktionen oder Anregungen von Projektexternen und -internen z.B. wegen ihrer qualitativ hochwertigen Beschreibungen und ihrer Anschlussfähigkeit in die Diskussion eingebracht werden. Die Projektmitglieder mit Reputation beeinflussen die Diskussion manchmal auch durch den Hinweis auf alte Probleme, die früher in den Mailinglistenarchiven diskutiert wurden. Ihre Informationsmacht zeigt sich eher in Mailinglisten mit begrenztem Schreibzugriff, wie „KDE-core-devel“, in der diese Personen mit Reputation über die zukünftige Entwicklung des Großprojekts diskutieren können. Obwohl die begrenzten Schreibzugriffe in neuen Mailinglisten wegen der üblicherweise vielen unwichtigen Emails eingeführt wurden, können bestimmte Problemsichten mit Einschränkungen vordiskutiert und „vorentschieden“ werden. Dies ist aber nur begrenzt möglich, da die Liste auch von anderen Mitgliedern gelesen wird und die Mitglieder dieser Mailinglisten ggs. auf Widerspruch stoßen können (Nachinterview EB).

Allgemein dürfen in Mailinglisten nur bestimmte Themen, meistens hinsichtlich der Softwareentwicklung des Subprojekts, besprochen werden. Verstöße dagegen und gegen die „Netiquette“ wie persönlich beleidigende Emails werden von allen Projektbeteiligten durch Ignorieren oder Zurechtweisungen bestraft. Das Ignorieren hat aber auch seine Grenzen, da Äußerungen von einzelnen Projektmitgliedern als Meinungsabbild des Projekts erscheinen und die Außendarstellung beeinträchtigen können (Gesprächsprotokoll SK). Auch der Rückzug aus einem kooperativen Verhältnis kann als Drohung dienen, wurde aber kaum berichtet. Lob oder Tadel vor Dritten ist aber im KDE-Projekt äußerst selten, was auf den Mailinglisten und während der Konferenzen festgestellt wurde. Folglich wird die verbale Bestrafung oder Anreizsetzung nicht strategisch verwendet (Gesprächsprotokoll NH; Mailinglistenauswertung1).

- *Wettbewerb* tritt zwischen zwei und mehr Parteien z.B. auf Mailinglisten oder im Chat-system auf, die bei einem Konflikt um die Zustimmung der anderen zu einer der vorgeschlagenen Alternativen werben. Wenn dieser seltene Fall auftritt, werden die Argumente verglichen und eine Alternative von oftmals wenigen Beteiligten ausgewählt. Schweigen wird dabei als Zustimmung gewertet, da davon ausgegangen wird, dass sich die Personen beteiligen, deren Interessen betroffen sind. Der Abstimmungsprozess im

Chatsystem erfolgt einfacher als in Mailinglisten, da dort alle anwesenden Beteiligten identifizierbar und zeitlich synchron anwesend sind. Dies vereinfacht die Abstimmung, ist aber auch nur bei einer geringen Anzahl von Personen effektiv. Vor allem Projektmitglieder aus dem inneren Kreis, aber auch Subprojektmitglieder kommunizieren im Chatsystem miteinander. Dadurch haben die Projektmitglieder aus dem inneren Kreis einen Vorteil gegenüber peripheren Personen, was sich auf eine Hierarchie/Macht auswirkt (s.o.). Eine offizielle Wahl mit Mehrheitsentscheid über die Homepage/Website erfolgte bei der Zieldefinition, aber auch bei anderen Teilprozessen nicht.

Zusätzlich existiert vor allem in diesem Teilprozess in manchen Mailinglisten ein Wettbewerb um Aufmerksamkeit, da viele für den Fortgang der Arbeit unwichtige Emails geschrieben werden. Die Mitarbeiter lesen dann oftmals die Emails, die von Personen mit Reputation geschickt werden (vgl. Brand, Holtgrewe 2007). Dies kommt auch in anderen Teilprozessen der Arbeitsplanung und der Leistungserstellung vor, wobei die Emails nach Bekanntheit der Absender gelesen werden. Das Vorhandensein dieses Wettbewerbs und seine Dämpfung zeigt sich in der Etablierung von Mailinglisten mit begrenztem Schreibzugriff (s.o.) auch für Subprojekte.

#### Arbeitsverteilung von Arbeitspaketen mit detaillierter Entwicklungsplanung

Eine detaillierte Entwicklungsplanung und die Arbeitsverteilung von Arbeitspaketen findet in den Subprojekten bei den Softwaremodulen/-teilprogrammen statt. Unter Arbeitsverteilung von kleineren Arbeitspaketen wird auch die Übernahme von größeren Aufgabenbereichen mit thematischem Softwarebezug verstanden. KDE-Mitarbeiter übernehmen eigenständig und freiwillig neue Arbeitspakete, Aufgaben bzw. Aufgabenbereiche, nachdem ein Problem, kleiner Wunsch bzw. Fehler durch andere oder den Softwareentwickler als Nutzer identifiziert wurde und in den Wissens-/Arbeitsbereich oder das Interesse des Softwareentwicklers fällt (vgl. für andere Projekte: Mockus et al. 2002). Die Akteure übernehmen unterschiedliche Aufgaben bzgl. ihrer Subprojekt-Software, um Doppelarbeiten zu vermeiden. Zugleich planen die Softwareentwickler ihr Vorgehen eigenständig. Oftmals besteht nur eine minimale detaillierte individuelle Entwicklungsplanung mit einem anvisierten groben Teilziel ohne Vorstellungen über den konkreten Ablauf (Interview EB).

Der Stellenwert der Koordinationsformen unterscheidet sich kaum vom Teilprozess Zieldefinition. Die Kooperation bestimmt durch häufig vorkommende Interdependenzen zwischen Softwarebereichen und der hin und wieder vorkommenden, freiwilligen Übernahme von Aufgaben diesen Teilprozess. Hierarchie/Macht ist von mittlerer Bedeutung, da die Übernahme von Aufgaben nicht erzwungen werden kann, aber die (Sub-)Projektleiter den Überblick über das Subprojekt behalten. Konflikte und ein Wettbewerb um die Qualität der Fehlerberichte, Wünsche und Kritik zwischen Projektmitarbeitern bestehen eher selten. Im Folgenden stehen die Unterschiede zwischen den Koordinationsformen im Fokus.

- Eine *Kooperation* zeigt sich bei der Übernahme eines Arbeitsbereichs bei Absprachen mit abschließendem Konsens. Die Übernahme von kleineren Arbeitspaketen wird oft einfach festgelegt und eine konsensuelle Zustimmung vorausgesetzt. Anhand folgender Aussagen lässt sich die Bandbreite der Kooperation darlegen:

Die Softwareentwicklung beinhaltet „No special process. The very first thing I try to do is to talk to other people about it, discuss ideas and implementation details (I love teamwork). So usually I bug other KDE people on IRC

:-) . Often we have endless discussions about apparently trivial things like how to name a certain method to make it a good API and the like.“ (Selbstaussage 45) und „If I have a new feature, I commit it when it's complete.“ (Selbstaussage Nr. 62).

Auf der KDE-Website, geben die einzelnen Projektleiter mit ihren Mitentwicklern manchmal, ab der Softwareversion 3.0 für jeden kommenden Release, ausdiskutierte Absichtserklärungen über weitere Entwicklungen in den Softwareprogrammen/-modulen dem Gesamtprojekt bzw. dem Releasekoordinator bekannt. Die Planung der einzubauenden Softwarefunktionen für den nächsten offiziellen Produktrelease erfolgt in den vorherigen Releaseperioden und wird am Anfang der freien Periode des Entwicklungsreleases festgelegt.<sup>102</sup>

Interdependenzen in der Software, z.B. wenn verschiedene Programme auf eine Bibliothek zurückgreifen oder Schnittstellen zwischen Programmen bestehen, geben der reziproken Kooperation auf Mailinglisten (innerhalb und zwischen Subprojekten) in den überwiegenden Fällen eine hervorgehobene Stellung. Dabei werden Übergänge zwischen Softwareteilen standardisiert oder neue Anforderungen ausgehandelt und im Teilprozess Leistungserstellung programmiert. Dies liegt nach meinem empirischen Datenmaterial an der objektorientierten C++-Programmiersprache und der gemeinsamen Qt-Bibliothek der Firma Trolltech, die das Anlegen von zentralen Bibliotheken fördern. Diese zentralen Bibliotheken werden von vielen anderen Softwaremodulen in Subprojekten verwendet. Gerade in Subprojekten mit mehreren Mitarbeitern, die sich auf ein gemeinsames Softwaremodul bzw. ein Programm beziehen, und vor allem bei zentraler und wichtiger Software in großen Subprojekten ist die reziproke Kooperation anzutreffen. Dabei können Veränderungen in einem Bereich Softwarefehler in anderen Softwarebereichen verursachen. Außerdem sind gegenseitige Absprachen gerade bei zeitweise auftretenden größeren Aufgaben im Subprojekt mit paralleler Bearbeitung von aufeinander aufbauenden Arbeitspaketen verbreitet. Dazu zählt auch das gegenseitige Aushelfen bei kleinen Aufgaben und die freiwillige temporäre Übernahme von kleinen Arbeitspaketen manchmal nach persönlicher Anfrage.

„I tend to look for things that no-one else is doing but are really in need of being done and just start working on it. However I also have little projects that I plan out for months and months before I start (if I ever start). I guess my process varies from time to time.“ (Selbstaussage 29).

Zusätzlich steigt die Wahrscheinlichkeit von Absprachen z.B. bei gemeinsamem Vorgehen, wenn die Projektbeteiligten sich kennen oder Reputation aufgebaut haben und somit von vornherein Vertrauen in die zukünftige Durchführung vorhanden ist. Vernachlässigt ein Projektbeteiligter in einem Subprojekt eine Aufgabe nach oder ohne Bekanntgabe, so wird diese in gruppenbezogener Kooperation übernommen, dies kann aber bei Wiederbeteiligung des Projektbeteiligten zu Konflikten führen (Interview EB; vgl. Brand, Holtgrewe 2007). Diese gruppenorientierte Kooperation, bei der die Zustimmung vorausgesetzt wird, wird im Folgenden deutlich: „It's great to see, when other people pick up projects you don't have time for anymore. That shows that open-source really works.“ (Selbstaussage 25). Die Übernahme von vernachlässigten Aufgaben erfolgt dabei laut Selbstaussage Nr. 34 fließend: „Later on, being interested in [a program], I contributed to [this program] and quickly became the main developer as the original author had still less time to spend on it.“ Eine weitere Arbeitsteilung innerhalb des Gesamtprojekts ermöglicht die Verteilung von Arbeitsaufgaben in den Subprojekten und eine gruppenorientierte Kooperation, die im folgenden Emailausschnitt anhand eines ungebetenen Vorschlags verdeutlicht wird:

„[Name1] wrote:

> I suggest changing the id of the select field to select\_location. If

> we do that atleast we will have one more happy camper on the dot.o thread..

I made this change. Really ought to have validated before the launch...

[Name2]“

---

<sup>102</sup> s.a.: <http://developer.kde.org/development-versions/>; <http://www.nationmaster.com/encyclopedia/KDE>

Fehlerberichte und Fehlerkorrekturen von Software werden häufig während der Leistungserstellung von Projektmitgliedern als gleichzeitige Nutzer dieser Software ohne Ansehen der Person dem anderen Projektmitglied oftmals freiwillig z.B. über das Fehlerberichtssystem, die Mailingliste oder selten über eine direkte Email übermittelt (vgl. andere Projekte: Iannacci 2003). Dies ist auch oft auf Mailinglisten oder in Chats der Fall, in denen die z.T. unbekanntes Mitlesenden zur Übernahme von Arbeitsaufgaben wie dem Testen von Software gebeten werden und sich ein Projektmitglied freiwillig meldet. Zusätzlich wird im Chatsystem häufig nicht nur über die persönliche Arbeit, sondern auch über Privates, Persönliches gesprochen oder Musikdateien ausgetauscht, wobei bei häufiger Nutzung des Chatsystems auch Freundschaften entstehen können. In Mailinglisten wird fast nur über die Softwareentwicklung kommuniziert, da Persönliches nicht an Unbekannte oder an ein großes Publikum gehen soll (Interview DF). Die Freundschaften gehen mit reziproker Kooperation einher, auf die bei Problemen zurückgegriffen werden kann. Es gibt „Springer“, meistens mit hoher Reputation, die unaufgefordert in diversen Subprojekten aushelfen. Gerade Ein-Personen-Projekte sind auf diese Solidarität im Großprojekt angewiesen (Konferenzprotokoll NH).<sup>103</sup>

- *Hierarchie/Macht* zeigt sich in diesem Teilprozess wie in allen Teilprozessen nicht durch physische Bestrafungsmöglichkeiten, da wegen der Virtualität und des Verlusts der intrinsischen Motivation die Übernahme von Aufgaben nicht erzwungen werden kann. Der Ausschluss aus Mailinglisten wird aus obigen Gründen nicht praktiziert und würde nur bei Personen wirken, die viel in die Arbeit im Projekt investiert haben. Es ist zwar möglich, Anerkennung zu erlangen, diese wird jedoch in Form von Reputation nur indirekt kommuniziert (vgl. für andere Projekte: Lerner, Tirole 2000: 14ff.). Dabei setzen nur wenige Projektleiter ihre Identifikationsmacht als charismatische Führer mittels besonders freundlicher und konstruktiver Email-Kommunikation und der Verknüpfung von allgemein geteilten Werten/ideologischen Zielen bei der Verteilung der Aufgaben erfolgreich ein. Manchmal überzeugen Projektleiter, aber auch Mitglieder mit hoher Reputation, durch ihr charismatisches Auftreten und die damit verbundene Anerkennung in Form von Aufmerksamkeit vor allem neue Mitarbeiter bestimmte Arbeitspakete in ihrem Subprojekt, auch größere Aufgabenbereiche oder Posten, zu übernehmen. Als weiterer belohnender Anreiz für eine kontinuierliche Anstrengung kann die bei beständiger Mitarbeit erfolgende, hin und wieder vorkommende Aufnahme in Mailinglisten mit begrenzter Teilnehmerschaft, wie kde-core-devel, Vergabe von KDE-Emailaccounts (@kde.org) oder in den Verein KDE e.V. gewertet werden (Interview MK; Konferenzprotokoll NH).

Der Projektleiter ist besonders verantwortlich für ein Projekt, d.h. er hat das Subprojekt zu verwalten, weswegen er bei seinen Subprojektmitarbeitern ein höheres Ansehen genießt. Er ist der Ansprechpartner für die anderen Subprojektmitarbeiter, wenn z.B. Probleme auftreten. Die Wichtigkeit des Subprojektleiters für dieses Subprojekt dokumentiert Selbstaussage Nr. 13:

„Lead, follow or get out of the way. It describes very good how [the project] developments works. Not in a macroscopic way. We don't have a leader that everybody follows, but in a microscopic sense. For every important part of [the project] there is somebody who cares for it. Many others help him or stay out of their way and work on other things. So there is usually little noise created by people who only talk but never do things.”

Auf der anderen Seite ist mit dem Projektleiterposten in größeren Subprojekten ein besonderes Vertrauensverhältnis und eine erhöhte Erwartungshaltung der restlichen Subprojektmitarbeiter hinsichtlich des Arbeitseinsatzes verbunden. Aus diesem Grund arbeitet der

---

<sup>103</sup> Die geringe Größe des Projekts hängt zum einen an dem Charakter des Bearbeiters oder zum anderen an der peripheren Ausrichtung der Software.

Projektleiter oft mehr als die anderen Projektbeteiligten an der Subprojektsoftware, übernimmt unangenehme oder unpopuläre Arbeitspakete/-bereiche oder überwacht die hohe Anzahl der ständig hereinkommenden Fehlerberichte. Damit ist der Projektleiter eher über neueste Entwicklungen informiert als seine Mitarbeiter. Manchmal ordnet der Projektleiter auch Arbeitspakete zu (Informationsmacht), d.h. er sendet einen Fehlerbericht an das entsprechende Subprojektmitglied, wenn in größeren Projekten die Aufteilung der Arbeitsbereiche erfolgte (Interview DF; Konferenzprotokoll LT1). Alle Aktionen von (Sub-)Projektleitern und Softwareentwicklern müssen aber letztendlich vor den anderen Mitarbeitern argumentativ bestehen, was eine Begrenzung der Hierarchie/Macht und ein Rückbezug auf die Kooperation ist.

- *Wettbewerb* zeigt sich beispielsweise in einer Motivationserhöhung. Die Projektbeteiligten wählen u.U. schwierigere und komplexe Arbeitspakete aus, um Reputation zu akkumulieren und einen zukünftigen Wettbewerbsvorsprung zu erhalten. Einen Wettbewerb bei der Übernahme eines Arbeitspakets aufgrund der Reaktionsgeschwindigkeit eines Projektbeteiligten gibt es nicht, da eine parallele Bearbeitung durch den offen zugänglichen Quellcode möglich ist (Interview EB).

Werden Fehlerberichte, Wünsche, Kritik etc. von Projektexternen und –internen eingesandt, vergleichen hier die Softwareentwickler, aber vor allem die Subprojektleiter, wenn sie Zeit haben, über die genügende Qualität, Verständlichkeit und Anschlussfähigkeit des Fehlerberichts oder der Anregung mit dem üblichen Standard im KDE-Projekt. Sie entscheiden dann über die weitere Bearbeitung, wobei auch die hohe Reputation eines Einsenders eine Rolle spielen kann. Da aber die Fehlerberichte und Wünsche normalerweise unabhängig von den Einsendern gesehen und die Einsendungen vom Fehlerberichtssystem standardisiert werden, ist Wettbewerb eher begrenzt (Interview RE).

Konflikte entstehen selten, gerade bei einer Kritik der Qualität oder Funktionalität von Programmen, die über die üblichen Fehlerrückmeldungen hinausgehen. Bei seltenen, schweren fachlichen Anschuldigungen zur Softwareausführung oder detaillierter Planung eines Subprojektarbeitsbereichs besonders von Projektbeteiligten wird eher mit Ironie und manchmal mit Gegenbeleidigungen als hierarchisch sanktionierende Elemente auf den Mailinglisten geantwortet, was in einen Schlagabtausch und Meinungswettstreit ausartet (Legitimationsmacht).

Beispielsweise kritisiert ein Endnutzer oder ein Softwareentwickler das Programm eines Softwareentwicklers oder die detaillierte Planung rüde, ohne sich den Quellcode anzusehen oder genug Wissen darüber zu haben. Dieser Softwareentwickler fühlt sich in seiner Ehre gekränkt und sieht die Norm der selbstorganisierten Softwareentwicklung angegriffen. Er schickt auch eine beleidigende Email zurück, was sich zu einem Hin und Her von beleidigenden Emails aufschauelt. Diese Art von Konflikten ist eigentlich nicht persönlich, da die Kontrahenten wenig über den anderen Wissen. „Die Art und Weise der Wortwahl entscheidet über Sympathie und Antipathie, nach dem Sprichwort: So wie es in den Wald hineinschallt so schallt es wieder hinaus.“ (Interview RN, EB).

Meistens enden solche Diskussionen mit der Feststellung, dass keine Einigung möglich ist. Dabei kann es vorkommen, dass eine Übernahme von Arbeitspaketen vorübergehend nicht erfolgt. Konflikte traten aber besonders bei der Mailingliste ohne Themeneinschränkung „KDE-cafe“ auf, da dort emotionsbeladene und politische Themen kontrovers diskutiert werden.<sup>104</sup> Diese Mailingliste wird nur von wenigen und nur sporadisch verwendet und

---

<sup>104</sup> In der Mailingliste KDE-cafe dürfen alle möglichen Themen, die nicht mit der Softwareentwicklung verbunden sind, besprochen werden. Diese Mailingliste soll für den persönlichen Kontakt zwischen den

von fast allen interviewten KDE-Mitgliedern abgelehnt (Interview RN, EB, DF; Bsp.-Emailthread 1, 2).

Einbindung der projektexternen Endnutzer in die Arbeitsverteilung der Arbeitspakete mit detaillierter Entwicklungsplanung

Die projektexternen Endnutzer beeinflussen die Zieldefinition oder die Arbeitsverteilung der Posten nicht. Sie beteiligen sich aber an der Arbeitsverteilung der Arbeitspakete mit detaillierter Entwicklungsplanung durch die Nutzung des Fehlerberichtssystems oder die Kommentierung in Blogs bzw. virtuellen Tagebüchern der Entwickler auf der Website/Homepage. Dabei melden sie, wie oben angesprochen, Fehler, formulieren aber auch Anregungen oder Wünsche, die von den Softwareentwicklern aufgegriffen werden können. Zusätzlich können sich Projektexterne an Entwickler in den Blogs/Online-Tagebüchern der Entwickler auf der KDE-Projekt-Homepage wenden (Interview DF). Dabei steht die Beziehung zwischen Softwareentwicklern und Endnutzern mit Koordinationsformen im Fokus.

- *Hierarchie/Macht* hat dabei eine hohe Bedeutung, da der Subprojektleiter oder seltener der entsprechende Softwareentwickler die Berichte und Wünsche oftmals sichtet und nach der Wichtigkeit einordnet, da Endnutzer alles als wichtig ansehen (Interview RE). Der Subprojektleiter leitet dabei in großen Projekten Fehlerberichte an bearbeitende Subprojektbeteiligte weiter.

- *Wettbewerb* ist Hierarchie/Macht nachgeordnet und von mittlerer Bedeutung, da die Fehlerberichte, Wünsche und damit indirekt die projektinternen und –externen Absender manchmal miteinander um Aufmerksamkeit konkurrieren. Der Subprojektleiter oder Softwareentwickler vergleicht diese nach Verständlichkeit, Anschlussfähigkeit und Reputation des Einsenders etc. mit im KDE-Projekt üblichen Standards. Dabei ist den Absendern als Endnutzer, von denen viele nur einen einzigen Fehlerbericht abgeben (s.o.), oftmals nicht bewusst, welche Kriterien angelegt werden oder dass ein Wettbewerb um Aufmerksamkeit zwischen vielen Fehlerberichten besteht.

- Weitere reziproke *Kooperationen* bestehen selten bei Verständnisproblemen von Fehlerberichten seitens des KDE-Softwareentwicklers oder Subprojektleiters. Diese versuchen, über Rückfragen bzgl. des gemeldeten Fehlers das Problem gemeinsam mit dem Fehlerberichtslegenden, z.T. externen Endnutzer zu lösen. Einmalige altruistische Kooperationen zwischen Projektbeteiligten und projektexternen Endnutzern haben eine geringe Bedeutung, da sehr viele projektexterne Endnutzer einmalig einen Fehlerbericht bzw. Wunsch über eine Programmfunktion abgeben. Die hohe Anzahl an einmaligen Berichten führt zu einer sehr hohen Gesamtzahl aller Beiträge, auch während der Leistungserstellung (Interview DF; Emailinterview 2).

Arbeitsverteilung durch Postenbesetzung

Zwischen Posteninhabern und den anderen Projektmitarbeitern besteht eine Arbeitsteilung, wobei die Posteninhaber das ihnen unterstellte gesamte (Sub-)Projekt verwalten und nach außen vertreten. Mit dem Posten ist eine Verantwortung für ein (Sub-)projekt oder einen großen Arbeitsbereich z.T. auch für längere Zeit verbunden. Die Posteninhaber fordern Mitarbeiter wie beschrieben auf, sich an die Normen der Mailinglisten- oder Versionsmanagementnutzung zu halten oder sind Ansprechpartner für (Sub-)Projektexterne. Im Folgenden stehen die Postenbesetzung und die dabei bestehenden Koordinationsformen im Zentrum, wobei die Postenbesetzung nicht nur für die eher informalen Projektleiterpos-

---

Beteiligten sorgen, hat aber eher zu Konflikten bzgl. der Meinung über andere Länder, ihre Bewohner oder die dortige Politik geführt (Interview RN, EB).

ten, sondern auch für die formalen Posten im Verein KDE e.V. gilt. Es wird dabei auf die Postenbesetzung fokussiert, bei der der bisherige, aktive Posteninhaber sich von einem wichtigen bzw. zentralen Posten z.B. Projektleiter zurückziehen möchte. Bei unwichtigeren Posten wie informalen Subprojektleiterposten, die einige Zeit verwaist sind, kann eine Person diesen Posten von sich aus übernehmen.

- *Hierarchie/Macht* geht vom bisherigen Posteninhaber aus, der im Allgemeinen, seit der Projektgründung einen Wechsel initiiert und die Übergabe aller Zugriffsrechte durchführt, d.h. er sucht immer seinen Nachfolger aus (eine besondere Art der „Kooptation“). Durch die geschickte Auswahl des Nachfolgers aus der Erfahrung der Softwareentwicklung und der Kenntnis der Mitarbeiter (Ressource soziales Kapital) kann die weitere Entwicklung des (Sub-)Projekts beeinflusst werden. „Manipulativ“ kann die gelegentliche freundlich-werbende charismatische Kommunikation mit Anerkennungsvergabe zur Überzeugung eines adäquaten Nachfolgers führen (vgl. für andere Projekte: Raymond 1998a, b).

- Die Entscheidung für einen Nachfolger wird als allgemeiner Konsens betrachtet, wenn keiner etwas gegen den Nachfolger einzuwenden hat. Bei der Besetzung von wichtigen projektinternen Posten ist vor einer Entscheidung oftmals eine reziproke *Kooperation* zu beobachten. Andere Projektbeteiligte aus dem inneren Kreis helfen dem bisherigen Posteninhaber gelegentlich bei der Suche nach Personen für diese Posten durch eigene, unterstützende Suche, Erkundigungen oder Vorschläge. Diese Hilfe wird benötigt, da die Besetzung wegen des je nach Posten hohen Arbeitsaufwands und hohen Erwartungsdrucks mögliche Nachfolger abschreckt und Kandidaten für die Nachfolge erst festgestellt werden müssen.<sup>105</sup> Übernimmt ein Mitarbeiter einen unwichtigeren, verwaisten Posten, geht er von einer Zustimmung als kooperatives Element aus. Manchmal weisen Personen mit hoher Reputation neue und periphere Projektmitglieder auf diese Posten hin und ermutigen eine Übernahme.

- Die Besetzung von Posten hängt mit Reputation zusammen. Die Wichtigkeit des zu übernehmenden Postens korrespondiert mit der Höhe der Reputation und der Erfahrung eines Mitarbeiters, weswegen der bisherige Posteninhaber die verschiedenen möglichen Nachfolger vergleicht. Ein *Wettbewerb* ist aber begrenzt und unbedeutend, da einerseits ein aktives Bewerben um einen Posten mit dem direkten Vergleich zweier Bewerber nicht nachgewiesen werden konnte und die Anfrage zur Übernahme eine Ehre darstellt. Andererseits hängt die Übernahme selbst mit der Entscheidung und dem persönlichen Zeitkontingent des Nachfolgers zusammen. Bei den Vereinsposten besteht zwar eine (politische) Mehrheitswahl, die ein wettbewerbliches Element darstellt, doch hat die Abstimmung eher den Charakter einer legitimierenden Bestätigung durch die Vereinsmitglieder, da die Entscheidung schon vorher stattfand (Interview RN).

Wird ein verwaister Posten übernommen, kann der vorherige Inhaber weiterhin seinen Anspruch darauf geltend machen. Selten ergibt sich ein Konflikt zwischen dem vorherigen Posteninhaber und seinem „Nachfolger“, wobei es auch zu gegenseitigen Quellcodelöschungen kommen kann. Da die Mitarbeiter überwiegend über IuK-Technik kommunizieren und keine weiteren Erklärungen für eine Abwesenheit herangezogen werden können, ist ungeklärt, ab welcher Zeitspanne eine Nachfolge gerechtfertigt ist. Der Konflikt wird dann wie oben beschrieben durch ein weiteres Subprojektmitglied als Vermittler gelöst (vgl. Brand, Holtgrewe 2007).

Die bisherigen Posteninhaber waren oft Gründer dieser Posten. Bisher gaben sie diese Posten wegen der erworbenen Reputation nicht auf, obwohl sie weniger für das Projekt

---

<sup>105</sup> So werden manche Posten, wie der des Releasekoordinators, nur temporär (ein halbes bis ein Jahr) übernommen, da dieser einen hohen Aufwand an Arbeit und Zeit mit sich bringt.

arbeiteten. Ein Wettbewerb zwischen den Personen aus dem inneren Kreis besteht wegen der überwiegend freundschaftlichen Verhältnisse und wiederholten persönlichen Treffen auf Linux-Konferenzen, Großprojekttreffen etc. eher unerschwerlich. Die nachfolgende Generation hat aber mittlerweile hohe Reputation akkumuliert und möchte auch wichtige Posten übernehmen, weswegen z.T. neue, unwichtigere Posten als Ausweichmöglichkeiten geschaffen wurden. Wettbewerb zwischen den Generationen (z.B. Gründern und später Eintretenden) wirkt sich auf die alltägliche Softwareentwicklung weitestgehend nicht aus. Eher erfolgt ein langsamer Ablöseprozess, bei dem der neuen Generation nach und nach die Posten überlassen werden (Nachinterview EB).

### 5.4.3.3. Leistungserstellung mit detaillierter Entwicklungsplanung

Bei der Leistungserstellung können Koordinationsformen nicht nur bei der Softwareentwicklung als Bearbeitung von Arbeitspaketen, sondern auch in der Beziehung zwischen Softwareentwicklern und Endnutzern festgestellt werden.

#### Koordination der Bearbeitung von Arbeitspaketen

Nachdem ein Arbeitspaket/Arbeitsbereich übernommen wurde, wird hauptsächlich nach dem Versuchs- und Irrtumsprozess bei der Bearbeitung von Arbeitspaketen vorgegangen. Dies belegt beispielsweise Selbstaussage 47: „I don't draw any structure plans. I simply start coding, and if there is somewhere a problem, then I continue somewhere else. A few days later I have often a better idea for the first problem.“ Dabei tritt die detaillierte Entwicklungsplanung sowohl mit der Arbeitsverteilung von Arbeitspaketen als auch bei der Leistungserstellung auf. Bei der zyklischen, parallelen Bearbeitung geht die Arbeitsverteilung von Arbeitspaketen der Leistungserstellung voraus. Die KDE-Mitarbeiter haben ein grobes Ziel vor Augen, das von Endnutzern, durch die eigene Nutzung oder durch Projektmitarbeiter an sie herangetragen und u.U. diskutiert wird. Bei der Leistungserstellung werden dann die Arbeitspakete normalerweise an einer lokalen Kopie z.B. im eigenen Haushalt bearbeitet (s.o.), wobei anhand der bisher produzierten Software eine besonders detaillierte, zeit- und problemnahe Planung erfolgt.

Bei der Leistungserstellung dominiert die Kooperation, da durch die ständige Beobachtung der Aktivitäten häufig unaufgefordert Hinweise geliefert werden oder durch überschneidende Arbeitsbereiche ständige Absprachen nötig sind. Eine weniger bedeutende Hierarchie/Macht erfolgt manchmal über textbasierte Sanktionen, selten über Quellcodelöschungen oder Schreibzugriffentzug fürs Versionsmanagementsystem, da ohne Legitimierung eine Abwanderung zu erwarten ist (Legitimationsmacht). Wettbewerb ist kaum bedeutend, da die Reputation nur über den Arbeitseinsatz bzw. die Softwareentwicklung zu erhalten ist, die dann von anderen in einem unerschwerlichen Vergleich bewertet wird. Ein direkter Vergleich zwischen geschriebenem Quellcode findet dabei selten statt. Dies wird im Weiteren beispielhaft dargestellt.

- *Kooperation* kommt ins Spiel, wenn Arbeitspakete zusammen erstellt werden oder Hilfe bei deren Bewältigung benötigt wird. Falls sich Arbeitsbereiche überschneiden (s.o.), gibt es ständig wiederholte gemeinsame Absprachen. Während dieser reziproken Kooperation übernimmt ein Beteiligter selten auch Arbeitspakete eines anderen Beteiligten, falls dieser diese nicht erledigen kann. Nach den gefundenen empirischen Ergebnissen ist dies am ehesten in Subprojekten der Fall. Manchmal bitten auch Projektmitglieder andere (vor allem beim Chatten), ihre Software zu kontrollieren bzw. zu testen. Dabei wird die Software auch gegenseitig kritisiert. Es bilden sich oftmals zwei Parteien, die das Für und Wider diskutieren, bis sich ein Konsens abzeichnet, wie mit dem geschriebenen Quellcode zu verfahren sei (Nachinterview SK; Interview RN).

Die gruppenbezogene Kooperation tritt oft auf, weil alle Projektbeteiligten ständig Mailinglisten, das Fehlerberichtssystem und Versionsmanagementsystem nutzen und damit neueste Veränderungen beobachten, vor allem des für sie relevanten Quellcodes (Interview EB; vgl. auch: Raymond 1998a, b). Mitarbeiter berichten während der Leistungserstellung Fehler über das Fehlerberichtssystem, die bei der Arbeitsverteilung der Arbeitspakete als Aufgaben auftreten. Dies ähnelt entfernt einem Peer-review, der Unterschied besteht jedoch darin, dass dabei kein Fehler einer Person zugeschrieben wird.<sup>106</sup> Dadurch werden oft unaufgefordert Hinweise auf Probleme bei der Softwareentwicklung per Email versendet. Zusätzlich werden manchmal Fehler aus der Fehlerliste ohne vorherige Absprache mit dem Subprojektleiter bearbeitet und als kleine Quellcodeänderungen in das Versionsmanagementsystem gestellt. Selbstaussage 52 dokumentiert dies wie folgt: „When it comes to KDE, the nicest thing is when somebody fixes bugs for me I introduced. However I really don't expect that, please don't get me wrong. I'm very pleased by a positive comment already.“ Projektmitglieder beantworten sich auch Anfragen in Mailinglisten oder Chaträumen über allgemeine Softwareentwicklungsprobleme. Dabei wird eine generelle, noch nicht oder schon implementierte Idee vorgestellt und im Rahmen von konstruktiver Kritik diskutiert, sowie nach Verbesserungsmöglichkeiten gesucht und -vorschläge unterbreitet, wie folgende Selbstaussagen zeigen:

„Not really - basically implementing something, talking about it with other people, rewriting, talking, etc, etc.“ und „Often it is an iterative process. I start out with a design I have in mind, do a first rough implementation of it, you discuss with other people about it or find some problems, modify the design to fit the needs and start from the beginning. I usually write the documentation at some point after the first or second iteration, because it helps you detect flaws in your design“ (Selbstaussagen 17 und 14).

Die Antwort kommt normalerweise recht schnell von beliebigen Projektbeteiligten, die gerade online sind und das nötige Wissen haben, z.B. weil sie an ähnlicher Software im gleichen Subprojekt arbeiten, was folgende Email zeigt:

```
„On Tuesday February [??], 200[?] [?:??:?], [Name1] wrote:  
> Hi, apparently there is a problem with the php when register_globals is off. you can see that you  
> have a menu on kate.kde.org (which is on a server with register_globals on) and www.kde.org/info  
> (which is on a server with register_globals off). Can somebody of the php gods look into this?  
$DOCUMENT_ROOT doesn't work with globals off. You have to use  
$_SERVER["DOCUMENT_ROOT"], which of course breaks the use of  
$DOCUMENT_ROOT inside double quotes.  
[Name2]“
```

Das nötige Wissen haben eher die Projektbeteiligten mit hoher Reputation, die je nach Schwere des Problems periphere Mitglieder unterstützen. Dabei kann u.U. die Erfahrung der Posteninhaber/Mitarbeiter mit hoher Reputation die konkrete Programmierung beeinflussen (vgl. für andere Projekte: Taubert 2006).

- Ein Projektleiter, seltener Projektbeteiligte mit hoher Reputation, können andere sanktionieren und *Hierarchie/Macht* ausüben, indem eingestellter Quellcode wieder entfernt wird. Der Projektleiter kontrolliert normalerweise stichprobenartig den eingestellten Quellcode, wenn z.B. die beschreibenden Kommentare (logs) der Veränderungen Verdacht erregen. Die Stichprobenkontrolle als Normalzustand gilt für Langzeitmitglieder des (Sub-)Projekts. Für neue Mitglieder gilt eine stufenweise Annäherung an diesen Zustand. Ein Projektleiter kontrolliert bei Eintritt eines neuen Mitglieds in sein (Sub-)Projekt erst jede Quellcodezeile. Einige Zeit nachdem das Neumitglied einen qualitativ hochwertigen Quellcode geliefert hat, werden nur noch die Veränderungskommentare

---

<sup>106</sup> Die Gabe ist dabei hinsichtlich des Gebers personalisiert, aber nicht hinsichtlich des Empfängers (Taubert 2006: 63ff.).

(logs) auf Verdächtiges kontrolliert. Nach einem weiteren Zeitraum ohne Probleme kontrolliert der Projektleiter nur noch sporadisch. Es hat sich Vertrauen zu der Person eingestellt, was wiederum ein kooperatives Element darstellt (Interview RN; Konferenzprotokoll NH).

Normalerweise werden bei geringfügigeren Normenübertretungen wie irrelevante, beleidigende und themenunspezifische Aussagen auf der Mailingliste textbasierte soziale Sanktionen verwendet (s.o.; Legitimationsmacht). Erstellt ein Mitglied mit hoher Reputation fehlerbehafteten Quellcode, so wird er eher durch Ironie von einem anderen sanktioniert.

Manchmal zwingen Projektleiter und seltener Projektbeteiligte mit hoher Reputation durch das Löschen von für schlecht befundenen, fehlerbehafteten Quellcode vor allem Neulinge, aber auch andere Mitglieder zu einer erhöhten Qualität der Software. Es werden Texte in der Software gelöscht (Verhinderungsmacht), die nicht technisch professioneller Natur sind bzw. keinen neutralen Standpunkt einnehmen und z.B. nationale Ideologien transportieren. Dies tritt jedoch selten auf, da nur wenige Personen gegen diese Regeln verstoßen.

Ein technischer Ausschluss aus dem Versionsmanagementsystem in Form des Entzugs der Schreibberechtigung kommt eher selten vor und kann nur von wenigen Posteninhabern mit Zugriff auf die Infrastruktur im Auftrag des Projekts erfolgen. Der Entzug wird aber in Fällen angewendet, in denen eine größere Menge des lagernden Quellcodes gegen die projektspezifischen Interessen verändert oder gelöscht wurde. Dabei spielt eine aus Unwissenheit falsche Nutzung der Werkzeuge des Versionsmanagementsystemzugriffs eine Rolle (Konferenzprotokoll NH; Interview MK, RE, DF). In einem Fall bestand ein Konflikt zwischen „moralischen“ Autorenrechten, da ein Projektmitglied seinen vollständigen Quellcodebereich löschte, und dem projektspezifischen Interesse am Quellcode, der deswegen wieder hergestellt wurde (weitere Details vgl. Brand, Holtgrewe 2007).

Auf jeden Fall muss eine Löschung oder ein Entzug der Schreibberechtigung argumentativ legitimiert werden und mit den Normen des KDE-Projekts übereinstimmen. Dabei wird versucht, dem Normübertreter, aber auch anderen, die Entscheidung verständlich zu machen. Unbegründetes Löschen kann zu einem Gefühl der Ungleichbehandlung führen. Diese argumentative Untermauerung ist ein Rückgriff auf ein kooperatives Element, damit kein Verlust der intrinsischen Motivation bei dem Projektbeteiligten und keine Abwanderung zu anderen Projekten als sanktionsbegrenzendes Element erfolgt. Löschungen aus persönlichen Gründen, z.B. beanspruchten zwei Subprojektmitglieder den Posten als Subprojektleiter und löschten die neuesten Quellcodeveränderungen, können nur geschlichtet werden. Dabei wird durch eine kooperative Diskussion ein Kompromiss gesucht (Konferenzprotokoll NH; vgl. Brand, Holtgrewe 2007). Die Leseberechtigung für das Versionsmanagementsystem wird aber aufgrund der Norm der ungehinderten Informationsweitergabe und u.U. auch der General Public License nicht entzogen, da keiner bei der projekthärenten Weitergabe der Software von Veränderungen der Software ausgeschlossen werden darf. Der Entzug der Leseberechtigung könnte Proteste in der Bewegung auslösen und den Input an Neumitgliedern senken.

- *Wettbewerb* entsteht durch den Vergleich des Arbeitseinsatzes bzw. eigenen Arbeitsschritts mit dem Fortschritt der anderen Subprojektbeteiligten. Es wird aber nicht gezielt verglichen, sondern der Vergleich ist ein Nebenprodukt der ständigen Nutzung der neuesten Entwicklungsversion und der sporadischen Beobachtung des Versionsmanagementsystems. Ein direkter Vergleich ist wegen der Komplexität der Software auch nur mit Mitarbeitern des gleichen oder von ähnlichen Subprojekten möglich. Eher selten findet auch ein direkter Vergleich nach der Qualität des eingeschickten Quellcodes im Subprojekt statt: Zwei Personen, meistens eine aus dem Subprojekt und eine von außerhalb des Subprojekts, senden zu einem Fehler fast zeitgleich je eine Korrektur. Selten wird für oder gegen bestimmten Quellcode oder Designs z.B. der Website/Homepage über eine Wahl

mit Mehrheitsentscheid über die Homepage entschieden. Eine Ausnahme zeigt der Ausschnitt aus einer Email zu Designänderungen: „So there won't be a vote for the default style? (Why not?) Or do you plan to do this later? (Why wait?) The styles are ready for the vote since a few weeks, we are only waiting for the maintainers to agree or disagree with Dre setting up the voting mechanism.”

Reputation wird zwar durch den Arbeitseinsatz bei der Softwareentwicklung erreicht, doch durch den Vergleich kann die eigene Position in der Reputationshierarchie festgestellt werden. Zusätzlich erhöht sich die Motivation über eine ständig hohe Qualität des Quellcodes bei ständiger Mitarbeit, einen Aufstieg in der Reputationshierarchie und die Übernahme eines Postens zu erreichen (vgl. für andere Projekte: Raymond 1998a, b; Working Group 2000: 15f.). Der soziale Status der Reputation wird aber nicht demonstriert, sondern bildet sich bei den anderen Projektbeteiligten, die auch vergleichend beobachten. Understatement/Bescheidenheit wird durch die eigene Abwertung und Aufwertung der anderen Beteiligten erwartet.<sup>107</sup> Der verhaltene, wechselseitige Vergleich läuft bei KDE-Entwicklertreffen über das Vorstellen neuer Softwareentwicklungen einerseits für eine vordergründige Absprache bzw. einen Ideenaustausch, aber andererseits auch, um hintergründig Anerkennung zu erhalten. Außerdem werden in inoffiziellen Gesprächen und Chats Informationen über Projektbeteiligte und ihre neuesten Errungenschaften ausgetauscht, was ein kooperatives Element ist (Konferenzprotokoll NH; Interview EB; Nachinterview SK).

Einbindung projektexterner Endnutzer in die Leistungserstellung mit detaillierter Entwicklungsplanung

Neben der Einbindung von Nutzern durch das Fehlerberichtssystem oder Blogs bzw. die virtuellen Tagebücher der Entwickler gibt es noch eine Usability-Mailingliste<sup>108</sup>, mit der die Nutzer in die Softwareentwicklung einbezogen werden können. Die Usability-Mailingliste wurde erst spät (2003/4) eingerichtet, da die softwaretechnische Benutzerfreundlichkeit für Endnutzer ein Problem darstellte. Die Softwareentwickler ignorierten bisher die Benutzerfreundlichkeit, obwohl sie selbst Nutzer sind, da sie die selbstgeschriebenen Funktionen kennen und keine intuitive Benutzerführung benötigen.

- Auch in der Mailingliste dominiert weiterhin *Hierarchie/Macht* wie beim Eintritt. Das Verhältnis zwischen Softwareentwicklern und Endnutzern ist einerseits durch die Norm der eigenständigen Softwareentwicklung, andererseits durch die Norm der Offenheit und Freundlichkeit gegenüber Endnutzern als „Fehlerberichter“ und potenzielle Mitarbeiter gekennzeichnet (vgl. Brand, Holtgrewe 2007). Dieser Normenkonflikt wird gelöst, indem die Entwickler nur ausführlich formulierte Ideen annehmen, die einen direkten, zeitnahen Anschluss zur jeweiligen Softwareentwicklung des jeweiligen Softwareentwicklers haben. Dabei gibt es keinen Unterschied zwischen einfachen, wie das Aussehen von Buttons auf dem Desktop, und komplexeren Vorschlägen. Allgemein gehaltene und unverständliche Vorschläge werden eher selten beantwortet oder beachtet, d.h. die Endnutzer müssen annähernd ähnliche fachliche Kenntnisse haben, um die gleiche Sprache mit den Entwicklern zu sprechen und anerkannt zu werden (Email-Interview Usability1).

- Auf dieser Mailingliste herrscht ein relativ hoher, aber nachgeordneter Meinungswettbewerb zwischen den Mailinglistenbeteiligten. Bisher ist es auf der Mailingliste unter den vielen Endnutzern üblich, dass andere Endnutzer von Varianten einer Funktion oder neuen

---

<sup>107</sup> Somit kann von einem direkten, demonstrativen Signal, vergleichbar mit dem Fahren eines sehr teuren Autos als Statussymbol, in dem Open Source-Projekt KDE nicht gesprochen werden.

<sup>108</sup> <http://mail.kde.org/mailman/listinfo/kde-usability/>

Funktionen überzeugt werden sollen. Dies führt zu unstrukturierten, z.T. nicht technischen Diskussionen, deren Ergebnisse gerade wegen der Unstrukturiertheit nicht in die Softwareentwicklung eingehen. Dabei besteht ein Wettbewerb der Endnutzer um die Aufmerksamkeit der Entwickler, der jedoch oft gegenüber anderen KDE-Mitarbeitern verloren wird.

- Die Kooperation hat in der Mailingliste eine geringe Bedeutung, da sich die Endnutzer aus altruistischer *Kooperation* an der Diskussion in der Mailingliste beteiligen. Eine gruppenorientierte Kooperation besteht wegen ihrer Nicht-Zugehörigkeit nicht. Ein weiteres kooperatives Element hat sich in der Usability-Liste etabliert, indem sich mittlerweile eine Gruppe von Usability-Experten zum einen aus Softwareentwicklern, die die erhöhte Benutzerfreundlichkeit als wichtig erkannt haben, und zum anderen aus Endnutzern mit hoher KDE-spezifischer technischer Kompetenz gebildet hat. Für diese Experten wurde die technische Hürde des Zugriffs auf die laufende Entwicklung im Versionsmanagementsystem durch ein neues technisches Instrument verringert, woraus eine seltene, direkte, reziproke Kooperation zwischen diesen Experten und wenigen, weiteren Softwareentwicklern erwuchs. Zusätzlich vermitteln diese Experten als „Intermediäre“ zwischen Entwicklern und Endnutzern und versuchen dabei, zwischen diesen ggf. eine reziproke Kooperation herzustellen, was selten gelingt. Sie haben in einigen Fällen Diskussionen mit wenigen projektinternen Softwareentwicklern über bestimmte Softwarebereiche/Programme und deren Probleme aus dem Blickwinkel der Benutzerfreundlichkeit angestoßen und moderiert. Außerdem konnten sie vereinzelt einige Softwareentwickler anregen, ihre Software einzubringen und von Endnutzern testen zu lassen (Email-Interview Usability2; Nachinterview DM).

#### 5.4.3.4. Arbeitszusammenführung

Die Arbeitszusammenführung zu einem offiziellen Release ist, so zeigen die empirischen Ergebnisse, mit einer Softwareendkontrolle bzw. –tests verbunden (s.a. Kap. 2.3.). Ein direkter Endnutzerbezug bzw. –einfluss bei der Arbeitszusammenführung und den Kontrollen besteht nicht.

Hierarchie/Macht ist am bedeutendsten, da der Releasekoordinator als Verantwortlicher den Ablauf des offiziellen Releases immer kontrolliert und durch sein fachliches Wissen und seine Erfahrung mitbestimmt. Dabei ist Hierarchie/Macht nur etwas stärker bedeutend als Kooperation, da keine Anweisungen erteilt werden können (s.o.), aber in vielen Fällen Absprachen über Vorgehen und weitere Hilfen nötig sind. Ein Aufmerksamkeitswettbewerb ist kurzzeitig vorhanden, aber kaum bedeutend. Kooperative Elemente dämpfen diesen Wettbewerb, können ihn aber vor allem kurz vor dem Release nicht verhindern. Auf die Koordinationsformen wird im Folgenden eingegangen.

- *Hierarchie/Macht* geht bei der Arbeitszusammenführung vom legitimierten Posten des Releasekoordinators aus. Der Releasekoordinator bestimmt offiziell seit der Version 2.0, Ende 2000, mit seiner Verhinderungsmacht die Erstellung jedes häufig erstellten, stabilen, offiziellen Releases aus den Arbeitspaketen der Softwareentwickler.<sup>109</sup> Die übernehmende Person dieses Postens wird über die Website/Homepage aber auch auf den Mailinglisten bekannt gegeben.<sup>110</sup> Inoffiziell hat ein KDE-Gründer diesen Posten schon bei den Versionen 1.0, 1.1 und 1.89 übernommen.<sup>111</sup>

---

<sup>109</sup> Der Releasekoordinator übernimmt die Zusammenführung der Arbeit in einen Release anstelle eines Gremiums bzw. des inneren Kreises oder eines gemeinsamen Konsenses aller Beteiligten, wie in anderen Projekten praktiziert (vgl. Erenkrantz, 2003).

<sup>110</sup> s. z.B. <http://developer.kde.org/development-versions/>

<sup>111</sup> s. release schedule: <http://en.wikipedia.org/wiki/KDE>

Der Releasekoordinator stellt immer den Releaseplan auf, überwacht dessen Einhaltung und testet die Software zu bestimmten Terminen, auf Anfrage und am Ende auf Fehlerfreiheit. Dabei ist er der Ansprechpartner für alle Mitglieder. Er löscht oft Quellcodeteile bei Übertretung des Releaseplans oder fehlender Fehlerfreiheit, wobei seine Macht nur auf den Release begrenzt ist.<sup>112</sup> Die Subprojektleiter sind für die Zusammenführung der Arbeitsleistungen in den Subprojekten und die Fehlerfreiheit einer Software verantwortlich und ihm daher untergeordnet. Sie haben dadurch im Hinblick auf der Fehlerbeseitigung eine abgeleitete „Macht“ bzw. Autorität über die anderen Projektbeteiligten und können ggf. Quellcode löschen (Interview RN; Telefoninterview SK, DM).

Bei einem Ausschluss von Quellcode spielt die Wichtigkeit bzw. Zentralität der Programme eine Rolle. So wird der Releasetermin bei schweren Fehlern in Software, die wichtig für die Funktionsfähigkeit des Endprodukts ist, manchmal z.B. bei Version 3.0, Anfang 2002 (um einige Wochen) nach hinten gerückt. Bisher wurden nur periphere und stark fehlerbehaftete Programme oder Programmteile aus Releases ausgeschlossen, wenn diese die nacheinander abfolgenden Tests nicht bestanden. Das Ausschlusskriterium ist das Auftreten kritischer Fehler bei der Nutzung (z.B. Systemabstürze). Ein kompletter Ausschluss eines Mitglieds durch Entzug der Dateiablagesystem-Zugangsberechtigung erfolgt sehr selten, wenn unabgesprochene Änderungen, rechtlich bedenkliche Designs oder ungetestete Software kurz vor dem Releasetermin eingespielt werden und gegen die Norm des Releaseplans verstoßen wird. Dies ist mit ein Grund, warum seit Version 3.0, Anfang 2002, in einem Zeitraum kurz vor dem Veröffentlichungstermin des Releases Personen aus dem inneren Kreis, z.T. auch Subprojektleiter, für jedes Subprojekt die Softwareveränderungen kontrollieren, testen und gegebenenfalls ablehnen. Damit übernehmen diese eine Kontrollfunktion des Releasekoordinators, auch um die Fehlerentstehung durch unbedachte Quellcodeveränderungen bei der Fehlerbeseitigung zu vermeiden und dem Releasekoordinator Arbeit abzunehmen (Interview EB, RN; Telefoninterview SK, DM).

Der Releasekoordinator koordiniert die Arbeitszusammenführung also durch fachliche Kompetenz und Erfahrung, die als Expertenmacht in Form von ständig auftretender Überzeugungsarbeit die Fehlerfreiheit und Termineinhaltung gewährleistet und das Einschätzen der Schwere eines Problems sowie der Wichtigkeit einer Software unterstützt. Anweisungen, Befehle oder Zwang kann der Releasekoordinator nicht einsetzen, da die Projektbeteiligten nur von seinem Organisationstalent abhängen. Bei ihm laufen ständig Informationen zusammen, die er in eine starke Informationsmacht umsetzen kann. Bei einigen Personen erhöht u.U. sein Charisma in Form von freundlichen, aber bestimmten Aussagen zusätzlich die Legitimität von Entscheidungen. Oft muss er jedoch als Schlichter bei Konflikten über die Qualität von Software zwischen zwei Subprojektbeteiligten fungieren. In seltenen Fällen, wenn keine Lösung gefunden wird, entscheidet der Releasekoordinator als letzte Instanz über Problem- und Konfliktlösungen oder setzt einen Vermittler ein (Telefoninterview SK, DM).

- *Kooperation* hat eine etwas geringere Bedeutung als Hierarchie/Macht, da Probleme normalerweise zwischen Releasekoordinator, den Subprojektleitern und anderen Mitgliedern der problematischen Softwareteile diskutiert werden und eine gemeinsame, konsensuelle Lösung gesucht wird. Dabei werden oft Absprachen über zukünftige Arbeiten getroffen (reziproke Kooperation). Dazu gehört die Aufgabenverteilung bei den Tests. Die Projektbeteiligten in den Subprojekten sprechen auch häufig ohne Releasekoordinator das weitere Vorgehen der Fehlerbeseitigung ab. Sie helfen sich z.B. gegenseitig bei Pro-

---

<sup>112</sup> Dies tritt auch in anderen Projekten auf (s. Erenkrantz, 2003; Jörgensen, 2001, 333f.).

grammen mit besonders vielen oder zeitkritischen Fehlern, oder Fehlerberichter und Fehlerbearbeiter diskutieren ein Problem (Telefoninterview SK, DM).

Dabei spielt das gemeinsame Interesse, ein stabiles, fehlerfreies Produkt zu erstellen, als Motivation eine große Rolle. Dieses gemeinsame Interesse äußert sich in einer gruppenbezogenen Kooperation, die bei Personen mit hoher Reputation besonders ausgeprägt ist und mit einer hohen Verantwortlichkeit bzw. einem Verantwortungsgefühl für das Gesamtprojekt einhergeht. Je mehr Reputation eine Person hat, desto eher wird sie nicht nur dem eigenen Subprojekt, sondern auch anderen Subprojekten helfen. Dies zeigt sich auch durch eine kleinere Gruppe von Projektmitarbeitern mit hoher Reputation, die den Releasekoordinator zuweilen unterstützt, indem diese ihm bei der Überwachung des Releaseplans und der Fehlerfreiheit der Software der Subprojekte helfen (Telefoninterview SK, DM).

- Nicht nur wegen der vielen Absprachen, auch wegen der vielen Probleme stellt sich ein *Wettbewerb* der Beteiligten um die begrenzte Aufmerksamkeit des Releasekoordinators um Hilfe bei Entscheidungen, Problemlösung und aktiver Mitarbeit bei der Fehlerbeseitigung ein. Der Releasekoordinator vergleicht die Intensität und Wichtigkeit des jeweiligen Hilferufs und wendet sich dem dringendsten zu. In einem kurzen Zeitraum vor dem Releasetermin steigt der Wettbewerb an, da fast alle Projektbeteiligten Probleme lösen möchten (Konferenzprotokoll NH; Telefoninterview SK). Hier unterstützen wie oben genannt die Projektmitarbeiter mit hoher Reputation den Releasekoordinator.

Ein weiterer Wettbewerb in Form eines Designwettbewerbs und eines Anreizinstruments wurde neuerdings (2004) von einem Unternehmer einer KDE-nahen Firma in einem Teilbereich eingeführt, um die Qualität des Softwaredesigns (Aussehen der Icons, Hintergrundbilder etc.) für einen Release zu heben. Das Preiskomitee bewertete nach ästhetischen Kriterien, wobei die Sieger T-Shirts, Festplatten und andere Hardware gewinnen konnten. Dies führte zu einer merklichen Verbesserung des Designs und der Arbeitsaktivitäten vor Ende des Wettbewerbs bzw. Releases (Emailinterview 3).

## 5.5. Zusammenfassung

Das untersuchte Open Source-Projekt KDE gehört wie andere Open Source-Projekte zur Open Source-Bewegung und produziert wie diese für die Bewegung, aber auch für andere, eine grafische Benutzeroberfläche mit Anwendungsprogrammen als Softwareprodukt. Dieses Softwareprodukt wird einerseits durch eine spezielle Softwarelizenz vor einer proprietären Aneignung bewahrt, andererseits soll das Projekt die Hochwertigkeit und Konkurrenzfähigkeit gegenüber proprietärer Software aufzeigen. Neben den anderen Open Source-Projekten gehören Vertretungsorganisationen, Infrastrukturprojekte, bewegungsnahe (Klein-) Unternehmen und Endnutzer als individuell Beteiligte zur Open Source-Bewegung und stellen eine Umgebung für das KDE-Projekt dar. Die „Großgruppe“ Open Source-Projekt KDE besteht aus recht homogenen individuellen Akteuren, die vor allem intrinsisch motiviert sind. Dabei strukturiert das Zentrum-Peripherie-Prinzip die Software bzw. Subprojektstruktur im KDE-Projekt und die IuK-technische Infrastruktur aus Mailinglisten, Chatsystem, Website und Versionsmanagementsystem.

Im KDE-Projekt wurde der Arbeitsprozess der Softwareentwicklung systematisch empirisch untersucht, d.h. die Ausprägungen und der Stellenwert der Koordinationsformen in den Teilprozessen der Softwareentwicklung. Wettbewerb, Hierarchie/Macht und Kooperation als Koordinationsformen konnten in dem Open Source-Projekt KDE nachgewiesen werden, weswegen sich die hier vorgenommene Unterscheidung von Koordinations- und Organisationsformen als sinnvoll erwiesen hat. Betrachtet man das Gesamtprojekt hinsichtlich des Gesamtarbeitsprozesses ohne den Endnutzerbezug und ohne Teilprozess-

perspektive, dominiert zwar die Kooperation, Hierarchie/Macht hat aber ebenfalls eine hohe Bedeutung<sup>113</sup>. Wettbewerb ist vorhanden, spielt aber eine untergeordnete Rolle.<sup>114</sup> Obwohl die Kooperation innerhalb des KDE-Projekts die Oberhand über die Softwareentwicklung hat, dominiert in zwei Teilprozessen Hierarchie/Macht. Einzig im Teilprozess Eintritt/Rekrutierung hat Wettbewerb eine mittlere Bedeutung, sonst eine sehr geringe Wertigkeit. Strategische Kooperation tritt nicht auf. Das Unterlassen von bestimmten Verhalten kann einerseits durch Angst vor Strafe, aber auch durch die Orientierung an einer Norm einer Gruppe (gruppenorientierte Kooperation) erfolgen. Es stellt aber kein Problem dar, da auf das auftretende Verhalten geachtet wurde. In den Teilprozessen der Softwareentwicklung unterscheiden sich also die Bedeutungen der Koordinationsformen (s. Tab. 5-1), aber auch der Ausprägungen.

Tab. 5-1: Bedeutung der Koordinationsformen in den Teilprozessen der Softwareentwicklung des Open Source-Projekts KDE, eigene Darstellung

Koordinationsformen	Teilprozesse der Softwareentwicklung					
	Eintritt/Rekrutierung*	Arbeitsplanung:			Leistungserstellung mit detaillierter Entwicklungsplanung	Arbeitszusammenführung
		Zieldefinition	Arbeitsverteilung von Arbeitspaketen mit detaillierter Entwicklungsplanung	Arbeitsverteilung durch Postenbesetzung*		
Kooperation	+	+++	+++	++	+++	++
Hierarchie	+++	++	++	+++	++	+++
Wettbewerb	++	+	+	+	+	+

„+++“ sehr hohe Bedeutung; „++“hohe Bedeutung“; „+“geringe Bedeutung“  
 \* Diese Teilprozesse sind den anderen Teilprozessen nachgeordnet (s. Kap.2.3.).

Bei *Eintritt/Rekrutierung*<sup>115</sup> spielt der Subprojektleiter eines Softwaremoduls/-programms mit seiner Positions- und Expertenmacht die bedeutendste Rolle, da er immer über die Aufnahme des Quellcodes eines potenziellen Mitglieds und über dessen Aufnahme ins KDE-Projekt durch Vergabe einer Schreibberechtigung bestimmt (Verhinderungsmacht). Dabei versuchen die potenziellen Mitglieder mit den offiziellen Mitgliedern über wiederholtes Einsenden von funktionstüchtigem Quellcode mitzuhalten, was einem wettbewerblichen Element nahe kommt und sich in einer selbsttätigen Einarbeitung widerspiegelt. Auch kooperative Elemente sind vorhanden, spielen aber als reziproke Kooperation eine untergeordnete Rolle, da eine selbsttätige Einarbeitung erwartet wird. Es besteht dabei eine ähnliche Koordinationsformenkombination wie bei der Beziehung zwischen Softwareentwicklern und Endnutzern in den Teilprozessen Arbeitsverteilung von Arbeitspaketen und Leistungserstellung (s.u.). Da die Beziehung zwischen Endnutzern und Softwareent-

<sup>113</sup> Dies zeigt sich deutlich, wenn der Teilprozess Eintritt/Rekrutierung als Voraussetzung für die Arbeit im Open Source-Projekt ausgeblendet wird (s. Kap. 2.3.).

<sup>114</sup> Demgegenüber dominiert bei den Übersetzern, so ein Interviewter, Hierarchie/Macht mit einer sternförmigen Kommunikationsstruktur. Der Übersetzungsteamleiter bestimmt sowohl die Zieldefinition, Arbeitsverteilung von Übersetzungen als auch die Arbeitszusammenführung, indem er das Übersetzte stichprobenartig kontrolliert und auf dem Versionsmanagementsystem ablegt. In der Leistungserstellung übersetzt jeder für sich, wobei kein kooperativer Austausch stattfindet.

<sup>115</sup> Dieser Teilprozess ist eine notwendige, aber nicht hinreichende Voraussetzung, dass die Softwareentwicklung erfolgt und daher den anderen Teilprozessen nachgeordnet (s. Kap. 2.3.).

wickeln im Vordergrund steht, zählt der Teilprozess Eintritt/Rekrutierung nicht vollständig zur Softwareentwicklung im KDE-Projekt.

Innerhalb des KDE-Projekts wird dagegen die *Zieldefinition* bei der Arbeitsplanung durch einen Diskurs mit Konsens und gruppenorientierter Kooperation bestimmt, wobei aus gemeinsamem Interesse und einer Identifikation mit dem Projekt über Ziele diskutiert wird. Personen mit hoher Reputation durch ihre fachliche Kompetenz nehmen hin und wieder über Expertenmacht Einfluss, können so die Übernahme eines Arguments zwar nicht erzwingen, die Weitergabe von Informationen jedoch steuern (Informationsmacht). Ein Wettbewerb zwischen verschiedenen Alternativen, der sich in einem Konflikt äußert, tritt nur selten auf. Ein Wettbewerb um Aufmerksamkeit auf den Mailinglisten wird durch Mailinglisten mit begrenztem Schreibzugriff gedämpft. Weitere Wettbewerbsinstrumente werden nicht verwendet.

Bei der empirischen Untersuchung ergab sich, dass der Teilprozess Arbeitsverteilung von Arbeitspaketen mit dem Teilprozess detaillierte Entwicklungsplanung verbunden ist. Dies ist aber auch bei Leistungserstellung und detaillierter Entwicklungsplanung der Fall. Während bei der Arbeitsverteilung von Arbeitspaketen das weitere zukünftige Vorgehen geplant wird, wird bei der Leistungserstellung anhand der bisher produzierten Software zeit- und problemnah entschieden.

Auch bei der *Arbeitsverteilung von Arbeitspaketen mit detaillierter Entwicklungsplanung*, die ein Teilprozess der Arbeitsplanung ist, dominiert die Kooperation, da durch Interdependenzen in der Software häufig die reziproke Kooperation auftritt. Ähnlich zur Zieldefinition ist die gruppenorientierte Kooperation durch die häufige Übermittlung von Fehlerberichten besonders wichtig. Hierarchie/Macht macht sich hin und wieder durch Identifikationsmacht oder Informationsmacht eines Subprojektleiters oder eines Mitglieds mit hoher Reputation bemerkbar, da diese die Übernahme von Aufgaben ähnlich wie bei der Zieldefinition nicht erzwingen können. Zusätzlich übernehmen diese gelegentlich aus Pflichtgefühl auch unangenehme und unpopuläre Aufgaben. Wettbewerb zeigt sich ähnlich zum vorigen Teilprozess eher selten in der Auswahl reputationserhöhender Aufgaben, beim Vergleich eingesandter Fehlerberichte nach ihrer Verwendungsfähigkeit und in Meinungsstreits.

Demgegenüber erfolgt bei der *Arbeitsverteilung durch Postenbesetzung*<sup>116</sup> der Arbeitsplanung die Entscheidung über die Zuweisung eines Mitglieds zu einem Posten durch den bisherigen Posteninhaber, d.h. hierarchisch, solange dieser den Posten durch aktive Mitarbeit ausfüllt. Die Entscheidung sollte aber nicht gegen einen allgemeinen Konsens verstoßen, d.h. es besteht quasi eine Vetomacht eines Mitglieds. Außerdem ist das Informieren über die Reputation möglicher Nachfolger kooperativ. Die wettbewerbliche Vorauswahl über die Reputation geeigneter Kandidaten ist unbedeutend, da sich die potenziellen Nachfolger nicht um einen Posten bewerben können. Außerdem sind Konflikte über eine Nachfolge selten.

Die Koordinationsformenkombination bei der *Leistungserstellung mit detaillierter Entwicklungsplanung* unterscheidet sich wenig von der Zieldefinition und Arbeitsverteilung von Arbeitspaketen. Auch dieser Teilprozess ist durch gruppenbezogene Kooperation bestimmt, da am häufigsten auf Anfrage unaufgefordert von einem nicht vorher bestimmten Mitglied Informationen zu Problemen und Detailfragen weitergegeben werden. Eine reziproke Kooperation mit gegenseitiger Hilfe z.B. durch Übernahme von Arbeitspaketen oder Testen von Software ist auch oft zu beobachten. Hierarchie/Macht zeigt sich bei der gele-

---

<sup>116</sup> Dieser Teilprozess ist ein gesonderter Teilbereich der Arbeitsverteilung von Arbeitspaketen und daher den anderen Teilprozessen nachgeordnet (s. Kap. 2.3.).

gentlichen Überprüfung neu erstellter Software durch den Projektleiter, der u.U. dann fehlerhaften Quellcode löscht. Außerdem können hin und wieder textbasierte Sanktionen in Form von Legitimationsmacht auftreten. Selten wird der Schreibzugriff zum Versionsmanagementsystem für normübertretende Projektmitglieder gesperrt. Die Norm der ungehinderten Informationsweitergabe im Zusammenhang mit der General Public License verhindert den Entzug einer Leseberechtigung als Sanktion. Äußerst selten kann zwischen zwei zeitgleich eingesandten Fehlerberichtigungen verglichen werden. Ein Wettbewerb um Arbeitseinsatz und beste Programme erfolgt dabei eher unterschwellig durch eine Motivation zur Programmierung, wobei sich die Softwareentwickler mit anderen Softwareentwicklern vergleichen.

Abweichend von den anderen Teilprozessen mit Ausnahme von Eintritt/Rekrutierung dominiert bei der *Arbeitszusammenführung* Hierarchie/Macht durch die bestimmenden legitimierten Posten des Releasekoordinators zusammen mit den Subprojektleitern. Der Releasekoordinator kann über seine Verhinderungsmacht fehlerbehaftete Softwareteile ausschließen und hat die Letztentscheidungsmacht bei Konflikten. Er steuert zusätzlich durch seine fachliche Kompetenz in Form der Expertenmacht und als zentraler Anlaufpunkt für Informationen (Informationsmacht) den Release. Die Kooperation ist nachgeordnet, aber weiterhin bedeutend, da ein gemeinsames Interesse besteht, ein stabiles, fehlerfreies Produkt zu erstellen. Des Weiteren treten häufig reziproke, direkte Kooperationen mit Absprachen zwischen Releasekoordinator, Subprojektleitern und Projektbeteiligten, zwischen Mitgliedern mit hoher Reputation und zwischen den normalen Subprojektbeteiligten auf. Wettbewerb um Aufmerksamkeit des Projektleiters spielt nur kurz vor Fertigstellung eine, wenn auch untergeordnete Rolle. Dabei wird Wettbewerb durch eine Kooperation mit weiteren Mitarbeitern mit hoher Reputation gedämpft.

Nachdem auf das Auftreten und die Bedeutung der Koordinationsformen eingegangen wurde, wird im Folgenden der Zusammenhang zwischen IuK-Technik und Koordinationsformen untersucht. Dabei wurde festgestellt, dass einerseits die IuK-technische Infrastruktur die Softwareentwicklung beeinflusst, unterstützt und überhaupt erst ermöglicht. Andererseits wird das Softwareprodukt und die IuK-technische Infrastruktur nach den eigenen Bedürfnissen geformt und angepasst. Dabei wird die statische, gekapselte Infrastruktur als Artefakt auf ein Produkt angewendet.

Die KDE-Softwareentwickler sind zugleich Nutzer der Software, wobei das durch die Nutzung erworbene Wissen sofort und unmittelbar wieder in die Softwareentwicklung integriert werden kann (unverzögerte Wissensspirale). Sonst muss weiteres Wissen angeeignet werden (verzögerte Wissensspirale), wobei auch auf das Wissen anderer KDE-Mitarbeiter zurückgegriffen werden kann. Zusätzlich können Endnutzer ihr Wissen und z.T. ihren Arbeitseinsatz durch die IuK-technische Offenheit in die Softwareentwicklung einbringen (s.u. Beziehung zwischen Softwareentwicklern und Endnutzern).

Aus den empirischen Ergebnissen kann herausgelesen werden, ob und inwiefern die Nutzung bzw. der Lese- und Schreibzugriff der IuK-technischen Infrastruktur mit Koordinationsformen zusammenhängt. Es besteht aber kein unmittelbarer determinierender Zusammenhang zwischen einzelnen Werkzeugen der IuK-technischen Infrastrukturtechnik und Koordinationsformen. Diese Werkzeuge unterstützen jedoch wie in der Technikdefinition (s. Kap. 2.2.) bestimmte Elemente der Koordinationsformen. Alle Koordinationsformen treten dabei weitestgehend bei allen Werkzeugen der IuK-technischen Infrastruktur auf,

auch da Kommunikation, Information und Produktion in der Infrastruktur nicht starr voneinander getrennt sind.<sup>117</sup>

Generell unterstützen Mailinglisten, Chatsystem, Foren auf der Website/Homepage und Kommentare im Versionsmanagementsystem kooperative Diskurse mit Konsens. Zugleich treten in Mailinglisten und im Chatsystem auch die textbasierten Machtinstrumente Ignorieren und Zurechtweisen als Sanktionen sowie Informations-, Identifikations- und Expertenmacht als manipulative Hierarchie/Macht auf. Gleiches gilt für Meinungswettbewerb/Konflikte oder Mehrheitswahlen.

Der Zugang zu den Werkzeugen kann für Hierarchie/Macht in Form von Bestrafen und Belohnen verwendet werden. Belohnungen und Bestrafungen werden durch einige Posteninhaber im Namen des KDE-Projekts durchgeführt, die Zugriff auf die Infrastruktur haben. Die Vergabe eines Schreibzugriffs erfolgt in Mailinglisten als Anreiz für normenkonformes Verhalten. Der Lesezugriff bleibt, bis auf eine Ausnahme, generell auch für Projektexterne offen. Wie oben schon beschrieben, zeigt sich eine bestrafende Sanktion durch das Löschen von Quellcode im Versionsmanagementsystem oder dem Entzug der Schreibberechtigung von Projektmitgliedern für das Versionsmanagementsystem. Die Leseberechtigung bleibt aus Transparenzgründen offen. Eine Homepageveränderung ist nur wenigen möglich. Der offene Lesezugriff und die oben erwähnten Belohnungen und Bestrafungen in besonderen Fällen werden durch die Norm des freien Informationsflusses im Open Source-Projekt geregelt. Weitere Normen, die an dieser Stelle nicht selbst, aber deren Wirkung festgestellt werden konnte, bestimmen also die geringe Verwendung von spezifischer IuK-technischer Infrastruktur. Wichtige technische Werkzeuge für die Durchführung von Wettbewerb in Open Source-Projekten wie automatisierte Wahlverfahren nach dem Mehrheitsprinzip wie im Gnome-Projekt oder automatisierte Vergleichs-/Matchinginstrumente für Anbieter und Nachfrager wie bei Online-Auktionen-/Marktplätzen/-Jobbörsen werden nicht verwendet. Dies kann mit ein Grund für den geringen Stellenwert des Wettbewerbs im KDE-Projekt sein.

Auch ein Zusammenhang zwischen IuK-technischer Infrastruktur und Teilprozessen der Softwareentwicklung zeichnet sich nicht unbedingt ab. Tendenziell werden Mailinglisten und Chatsystem eher bei der Zieldefinition verwendet. Ansonsten kann jedoch kaum ein Bedeutungsunterschied zwischen den Teilprozessen festgestellt werden.

Wie bei der Wissensspirale angesprochen, werden auch projektexterne individuelle Endnutzer in die Arbeit wenigstens z.T. eingebunden und treten mit Softwareentwicklern in Kontakt. Dabei können sich die Endnutzer direkt an der Softwareentwicklung mit ihrer detaillierten Entwicklungsplanung und damit auch indirekt an Arbeitsverteilung und Leistungserstellung beteiligen. Jedoch haben Endnutzer auf Zieldefinition, Postenbesetzung und Arbeitszusammenführung keinen Einfluss, in denen für den Erhalt des KDE-Projekts zentrale Entscheidungen gefällt werden. Der Teilprozess Eintritt/Rekrutierung beruht dabei auf der vorherigen Beteiligung an der Softwareentwicklung, wobei potenzielle Mitarbeiter durch eine eigenständige Einarbeitung die fließende, halbdurchlässige Grenze des KDE-Projekts überwinden können. Dieser Zusammenhang bei der Beziehung zwischen Endnutzern und Softwareentwicklern in den genannten Teilprozessen zeigt sich in einer glei-

---

<sup>117</sup> Beispielsweise kann über die Homepage auf das Versionsmanagementsystem anonym zugegriffen werden. Neue Informationen werden auf der journalistischen Seite über die Mailingliste geschickt, die kommentiert werden können. Außerdem kann eine Kommunikation auch über die Kommentare in dem Versionsmanagementsystem erfolgen oder es können Quellcodezeilen über die Mailingliste verschickt werden.

chen, nachfolgend dargestellten Koordinationsformenkombination. Hilfen und Verbesserungsvorschläge werden gern gesehen, doch hängt die direkte Integration von Quellcode oder Vorschlägen (Fehlerberichten, Wünschen) von deren unmittelbarer thematischer und zeitlicher Anschlussfähigkeit zur gerade ablaufenden Softwareentwicklung ab (vgl. Brand, Holtgrewe 2007). Endnutzern mangelt es dabei an Reputation und den Ressourcen für Experten-/Informationsmacht, die sich in eine dominierende Hierarchie/Macht der Softwareentwickler umsetzt. Quellcode oder Vorschläge werden eingesendet und nach Fehlerfreiheit, Anschlussfähigkeit und Aufmerksamkeit mit dem projektüblichen Standard verglichen, was zu einem Wettbewerb mittlerer Bedeutung führt. Zusätzlich besteht eine Norm der selbstorganisierten Einarbeitung, weswegen Kooperation zwar auftreten kann, aber von geringer Bedeutung ist. Ohne diese Grenze gegenüber den Endnutzern, die über die Koordinationsformen nachgezeichnet wurde, wäre keine Konzentration auf die Softwareentwicklung möglich, da nicht programmiert und nur über mögliche Funktionalitäten debattiert würde. Durch ihre Durchlässigkeit können aber neue Ideen und potenzielle Mitarbeiter aufgenommen werden.

Im Open Source-Projekt KDE dominiert zwar Kooperation, aber ohne Hierarchie/Macht funktioniert dieses Arbeitsnetz nicht. Aufgrund der hohen Virtualität unterstützt die IuK-technische Infrastruktur alle Koordinationsformen, lässt aber nur spezifische Elemente zu. Wettbewerb herrscht unterschwellig über Reputation, die in der Leistungserstellung generiert wird, was erklärt, warum Wettbewerb kaum eine Rolle spielt. Dabei beeinflussen neben der General Public License und den autonomen Mitarbeitern projektumfassende Normen wie selbstorganisiertes Arbeiten und freie, ungehinderte, transparente Informationsweitergabe die Koordination.

## **6. Analytik der Koordinationsformen im Open Source-Projekt KDE anhand von Institutionentheorien**

In diesem Kapitel wird an die empirischen Erkenntnisse über die Koordinationsformen in den Teilprozessen des Open Source-Projekts KDE angeknüpft. Es werden Vermutungen theoretisch hypothetischer Art erarbeitet, die die empirischen Ergebnisse zu Ausprägung und Bedeutung der Koordinationsformen im KDE-Projekt u.U. als Hypothesen erklären können. Es wird vorsichtig von Vermutungen gesprochen, da bisher keine dominierende und einheitliche Theorie der Koordination in Organisationen existiert, anhand derer überprüfbare Hypothesen gebildet werden könnten. Ausgehend von diesen Vermutungen können dann abschließende und belastbare Erklärungen entwickelt werden. Eine Theorie der Koordination in Organisationen kann für das KDE-Projekt als Arbeitsnetz beispielsweise nicht aus dem naheliegenden Netzwerkdiskurs entwickelt werden. Wie schon die Aufarbeitung des Netzwerkdiskurses im Zusammenhang mit dem Diskurs über Organisationsformen in Kapitel 3.1. zeigte, besteht in diesen Diskursen noch keine einheitliche Begrifflichkeit zu Koordination und Koordinationsform sowie Organisation und Organisationsform. Deren Merkmale wurden zwar plausibel, jedoch heuristisch und eklektisch zusammengestellt, Querbezüge bestehen nicht. Eine Basis für eine Theorie der Koordination in Organisationen kann in der obigen Neudefinition dieser Begriffe gesehen werden. Aus dieser Neudefinition, aber auch aus dem Netzwerkdiskurs, konnten keine Begründungszusammenhänge dahingehend ausgearbeitet werden, wann welche Koordinationsform als Institution im Netzwerk eine Rolle spielt. Eine Ausnahme stellt die Neue Institutionenökonomie dar, die einen Ausgangspunkt für den Netzwerkdiskurs bildet (s.u.).

Zusätzlich wurden Handlungen im Arbeitsprozess des KDE-Projekts untersucht, was eine Betrachtung auch von individuellen Akteuren nahe legt. Damit kann das Auftreten von Institutionen anhand von Annahmen über das Akteursverhalten bestimmt werden. Da der Netzwerkdiskurs sich auf eine organisationelle Ebene bezieht, sind in diesem keine Vorstellungen über das Verhalten der Akteure vorhanden, aus denen Vermutungen über Normen entwickelt werden können.

Weitere, adäquate Theorien, die Organisationen betrachten, sind daher heranzuziehen, in denen Institutionen wie Koordinationsformen auftreten. Die Neue Institutionenökonomie wurde ausgewählt, da die Frage der transaktionskostentheoretischen Ausrichtung nach einer effizienteren Verwendung von Markt oder Unternehmen auch eine Frage nach der Koordination beinhaltet (vgl. Göbel 2002). Zusätzlich basieren die Koordinationsformen auf den Vorarbeiten dieses Theoriekonzepts (s. Kap. 3). Markt, Netzwerk und Unternehmen als Organisationsformen spielen bei der Transaktionskostentheorie und Hierarchie bei der Prinzipal-Agent-Theorie eine Rolle. Zudem kann auf eine systematisch durchgeführte Analyse der drei Koordinationsformen im elektronischen Arbeitsmarkt anhand der Neuen Institutionenökonomie hingewiesen werden (Becher et al. 2007).

Zusätzlich basiert die Neue Institutionenökonomie auf der Akteursvorstellung des Homo Oeconomicus (bzw. Rational Choice) mit instrumentellen Kosten-Nutzen-Kalkülen. Der Homo Oeconomicus ist nach Schimank (2000) neben dem Homo Sociologicus mit seinem normorientierten Handeln eine Akteursvorstellung, die Ansätze für eine Erklärung von Institutionen bietet. Beide Akteursvorstellungen als sehr verbreitete idealtypische Vorstellungen verstehen Institutionen entweder als Resultat strategischer Entscheidungen der Homines Oeconomici für oder gegen eine Regel oder als eine Ausrichtung der Homines Sociologici an vorhandenen sozialen Regeln. Sie sind dabei mit „Organisationstheorien“ wie der Neuen Institutionenökonomie verbunden, die sich auf die untersuchten Institutionen auf der Handlungs- und Mesoebene beziehen.

Neben der genannten Neuen Institutionenökonomie mit der genannten Akteursvorstellung des Homo Oeconomicus besteht der Neo-Institutionalismus mit der Akteursvorstellung des Homo Sociologicus. Weder die Neue Institutionenökonomie noch der Neo-Institutionalismus stellen jedoch abgeschlossene Theorien dar, sondern können eher als heterogene Theoriekonzepte, -gebäude bzw. theoretische Forschungsprogramme charakterisiert werden. Dies zeigt sich auch darin, dass die ausgewählten Theoriekonzepte nicht vollkommen mit den strikten Akteursvorstellungen übereinstimmen, sondern mehr oder weniger realitätsnahe Vorstellungen integriert haben und von den idealtypischen Vorstellungen abgerückt sind. Weiterhin zentral ist aber in der Neuen Institutionenökonomie das Rationalkalkül und beim Neo-Institutionalismus die Sozialisation von Regeln (Göbel 2002: 22ff.; Walgenbach 1999, 2002; Edeling 1999).

Betrachtet man die Neue Institutionenökonomie, werden in diesem Theoriekonzept zwar vor allem Profit-Unternehmen, aber u.a. auch Non-Profit-Organisationsformen untersucht, weswegen Profit-Unternehmensnetzwerke, aber auch Non-Profit-Arbeitsnetze wie Open Source-Projekte untersucht werden können (vgl. Theuvsen 1999). Die Neue Institutionenökonomie ist zu modifizieren, um Non-Profit-Organisationsformen detaillierter und Koordinationsformen überhaupt fixieren zu können. Doch ist die Erklärungsmöglichkeit der Neuen Institutionenökonomie begrenzt.<sup>118</sup>

Das Marktmodell wird in der Neuen Institutionenökonomie implizit als Bezugspunkt zur Erklärung von Unternehmen, Netzwerken, staatlich-bürokratischen Organisationen etc. herangezogen, was den untersuchten Phänomenen nicht vollständig gerecht wird. Dabei wird oftmals der Nutzenbegriff zu offen definiert, so dass dieser tautologisch wird (vgl. Pirker 2000: 72ff.; Göbel 2002: 22ff.).<sup>119</sup> Vor allem die Integration des Altruismus, nicht-kalkulativen Vertrauens oder anderer kooperativer Elemente fällt in der Neuen Institutionenökonomie schwer, da diese nur z.T. mit dem rationalen Akteur zu vereinbaren ist (vgl. Williamson 1993; Göbel 2002: 123; Noteboom 2002: 13).<sup>120</sup> Kooperation kann zwar aus einer instrumentellen Perspektive mit individuell-utilitaristischen Handlungsorientierungen (s. Austauschtheorien) betrachtet werden, diese Perspektive beleuchtet aber nur einen Ausschnitt des Phänomens Kooperation. Aus der Sicht der Sozialpsychologie ist Kooperation zusätzlich mit prosozialem Handeln verbunden, das in einen Altruismus gipfelt (s. Kap. 3.2.3.4.; vgl. Böhle, Bolte 2002: 149f.; Bierhoff, Klein 1992; Spieß 1998).

Weiterhin wird im folgenden Kapitel gezeigt, dass die Neue Institutionenökonomie Erklärungsmuster für die unterschiedliche Bedeutung der Koordinationsformen in den Teilprozessen des KDE-Projekts liefern, aber nach der Fragestellung dieser Arbeit keine Vermutungen über die (Herkunft der) Ausprägungen aufstellen kann.

---

<sup>118</sup> Die Begrenztheit der Neuen Institutionenökonomie erkennt auch Simon (1991: 41f.), der nur geringe Transaktionskostenunterschiede zwischen institutionellen Arrangements (d.h. „Meta“-Organisationsformen) sieht und daher eher historische und außerökonomische Faktoren für die Institutionenwahl als wichtig erachtet.

<sup>119</sup> Der Nutzenbegriff wird manchmal stark erweitert, so dass die Nutzenänderung anderer Akteure in der Nutzenkalkulation eines Akteurs inbegriffen ist oder jedes erklärbare, zielentsprechende Verhalten darunter gefasst werden kann. Dadurch läuft die Theorie aber Gefahr, sich zu immunisieren, da jedes Verhalten als nutzenerhöhend angesehen werden kann (für eine ausführliche Auseinandersetzung s. Göbel 2002: 25ff.). In dieser Forschungsarbeit wird von einem egoistischen rationalen Akteur ausgegangen, der seinen Nutzen ohne Betrachtung des Nutzens anderer und nur zur eigenen materiellen Wohlfahrtsmehrung erhöht. Der Nutzen eines Akteurs kann zwar durch andere verändert werden, doch wird dabei die Nutzenveränderung bei anderen nicht mitkalkuliert.

<sup>120</sup> Neuere Entwicklungen versuchen, Vertrauen und Reziprozität als kooperative Elemente zu integrieren (s.u.).

Diese Begrenzungen führten zur Suche nach einer weiteren Institutionentheorie mit Akteursvorstellung und Organisationsbezug, wobei sich der Neo-Institutionalismus als implizite Netzwerktheorie anbot. Das Theoriekonzept Neo-Institutionalismus wurde ausgewählt, da es wissenschaftshistorisch als eine Gegenbewegung zur Neuen Institutionenökonomie konzipiert worden ist und die gleichen Phänomene bei ähnlichen bzw. gleichen Untersuchungsobjekten eher aus organisationalen und gesellschaftlichen (Netzwerk-)Zusammenhängen zu erklären versucht (s. Kap. 6.2.; vgl. Hasse, Krücken 1999: 7ff.; Fligstein 1985, 1991). Damit können die Ausprägungen und Bedeutung der Koordinationsformen sowie besonders ihre Herkunft in den neu entstandenen Open Source-Projekten aus anderen gesellschaftlichen Zusammenhängen bestimmt werden. Als Gegenbewegung zur Neuen Institutionenökonomie konzipiert, betrachten Autoren des Neo-Institutionalismus auch Organisationen, die durch den anderen Theorieansatz untersucht wurden. Folglich kann der Neo-Institutionalismus auch die Bedeutung der Koordinationsformen betrachten. Die Ausprägungen und Bedeutung der Koordinationsformen kann aber nur auf einer eher abstrakten gesellschaftlichen Ebene fixiert werden, da Literatur herangezogen wird, die die einflussnehmenden Gesellschaftsbereiche als Ganzes fokussiert. Die Ausprägungen und Bedeutung der Koordinationsformen in den Teilprozessen werden auch untersucht, doch können an dieser Stelle nur sehr schwache Bezüge angedeutet werden, da es an detaillierter, empirischer Literatur zu den Teilprozessen in den einflussnehmenden Gesellschaftsbereichen mangelt.

Mit diesem Theoriekonzept wurden Non-Profit-Organisationen wie Museen, Schulen etc. untersucht, weswegen es sich für die Analyse eines Open Source-Projekts geradezu anbietet (vgl. Hasse, Krücken 1999; Walgenbach 1999). Außerdem kann der Neo-Institutionalismus Kooperation mit Vertrauen einfacher integrieren, da nicht-kalkulatives Vertrauen oder Kooperation im Zusammenhang mit bestehenden gesellschaftlichen Normen oder Vorstellungen eingebunden werden kann. Bei nicht-kalkulativem Vertrauen z.B. basiert das Vertrauen auf bestimmten Verhaltensweisen in Form von Signalen, die nur im Zusammenhang mit bestehenden Normen oder Vorstellungen einen Sinn ergeben (s. Kap. 3, 6.2.). Der Neo-Institutionalismus ist auch deswegen für eine Erklärung geeignet, weil sich in der Kultur als Werte- und Normensystem kooperative, wettbewerbliche und hierarchische Elemente wiederfinden lassen, die als gesellschaftliche Zusammenhänge auf das individuelle Verhalten einwirken können (s. Cultural Theory, Thompson et al. 1990). Spezifische Koordinationsregeln bzw. -mechanismen gehen dabei durch die gesellschaftliche (Netzwerk-)Einbettung aus der spezifisch beeinflussenden gesellschaftlichen Umgebung in die zu untersuchende Organisation ein. Dabei gehorcht ein Individuum nicht automatisch allen Normen und Institutionen. Es bestehen manchmal konfligierende Institutionen, die Entscheidungsfreiheiten und damit eine „Deinstitutionalisierung“ ermöglichen. Bei spezifischen Institutionen bestehen keine konfligierenden Institutionen, so dass diese in bestimmten sozialen Situationen fest angewendet werden. Nur bestimmte Vorstellungen aus der gesellschaftlichen Umwelt treten in Organisationen auf, die mit der Umwelt in Gleichgewicht stehen. Da der Neo-Institutionalismus von dem Modell des Homo Sociologicus ausgeht, ist dieses Theoriekonzept zu modifizieren, um für diese Forschung anwendbar zu werden (s. Kap. 6.2.).

Beide Theoriekonzepte ergänzen sich also bei der Untersuchung der Koordinationsformen als Institutionen z.B. in einem Open Source-Projekt, da beide sich ergänzende theoretische Grenzen besitzen. Die Neue Institutionenökonomie bezieht sich eher auf die Handlungsebene und organisationelle Ebene<sup>121</sup>, wobei über den betriebenen (Ressourcen-

---

<sup>121</sup> Die Neue Institutionenökonomie wird zwar auch für Phänomene auf der Makroebene verwendet, die Theorie bezieht sich aber vornehmlich auf Unternehmen (vgl. Göbel 2002; Walgenbach 1999, 2002).

)Aufwand die Koordinationsformen und ihre Bedeutung erklärt werden können. Der Neo-Institutionalismus blickt, obwohl der Ausgangspunkt ein individueller Akteur in seinen gesellschaftlichen Bezügen ist, eher auf die Meso- und Makroebene, wobei über den gesellschaftlichen Kontext Begründungszusammenhänge zu den Koordinationsformen gefunden werden können.

Für eine Analyse des Untersuchungsobjekts Open Source-Projekt KDE sind beide Theoriekonzepte mit ihren realitätsnahen Vorstellungen an dieses Projekt anzupassen. Die Unterkapitel zur Neuen Institutionenökonomie (s. Kap. 6.1.) und zum Neo-Institutionalismus (s. Kap. 6.2.) sind dabei ähnlich aufgebaut.

Zuerst wird das Theoriekonzept derart modifiziert und interpretiert, dass es für eine Untersuchung des KDE-Projekts genutzt werden kann. Hintergrund dieser jeweiligen Theorie-modifikation ist die Frage, inwiefern die Neue Institutionenökonomie und der Neo-Institutionalismus für eine Betrachtung der Koordinationsformen und damit für eine Theorie der Koordination in Organisationen verwendet werden können.

Die vorgefundenen Koordinationsformen in den Teilprozessen werden im Anschluss anhand der modifizierten Theorie analysiert. Ausgehend von den empirischen Erkenntnissen werden Vermutungen entwickelt, die Begründungszusammenhänge für diese Erkenntnisse darstellen können. Dabei wird nur die Koordination der Teilprozesse der Softwareentwicklung beleuchtet, die Beziehung der Softwareentwickler zu den Endnutzern jedoch ausgeblendet. Auch weitere empirische Daten, die im empirischen Teil nicht erwähnt wurden, werden u.U. herangezogen, um Vermutungen zu unterstreichen.

## **6.1. Analyse der Koordinationsformen anhand der Neuen Institutionenökonomie**

Transaktionskostentheorie als Ausgangspunkt der Theorieinterpretation

Allgemein wird das Theoriekonzept Neue Institutionenökonomie in drei Bereiche mit unterschiedlichen Blickrichtungen und theoretischen Unterschieden wie Prinzipal-Agent-Theorie, Property-Rights-Theorie und Transaktionskostentheorie unterteilt (vgl. Göbel 2002; Richter, Furubotn 1996), was im folgenden Zitat prägnant widergegeben wird.

„Die Neue Institutionenökonomie stellt heute kein einheitliches Theoriegebäude dar. Vielmehr besteht sie aus mehreren methodologisch verwandten Ansätzen, die sich gegenseitig überlappen, ergänzen und teilweise aufeinander beziehen, teilweise aber auch unterscheiden“ (Picot et al. 2002: 54).

Dabei untersucht die Prinzipal-Agent-Theorie möglichst effiziente hierarchische Beziehungen z.B. in Unternehmen bzw. Auftraggeber-/Auftragnehmer-Beziehungen, die Property-Rights-Theorie die effiziente Verteilung und Gestaltung von Verfügungsrechten und die Transaktionskostentheorie Leistungsbeziehungen zwischen ökonomischen Akteuren im Allgemeinen und die Wahl bzw. Gestaltung einer effizienten Organisationsform im Speziellen (z.B. bei Make-or-Buy-Entscheidungen).

Da die Neue Institutionenökonomie ein heterogenes Theoriegebäude bzw. ein theoretisches Forschungsprogramm darstellt, wird im Weiteren eine Interpretation bzw. Modifizierung der Theorie angestrebt, die sich an der Theoriemodifizierung der Neuen Institutionenökonomie von Becher et al (2005) orientiert. Aus diesem Theoriegebäude sticht vor allem die Transaktionskostentheorie heraus (s.u.) und fungiert als Ausgangspunkt einer Theorieinterpretation/-modifizierung. Grob umrissen, untersucht die Transaktionskostentheorie die Kostenentstehung durch systemimmanente Markt- bzw. Transaktionsunvollkommenheiten. Dabei wird in der Theorie eine Begründung einer transaktionskosteneffizienten Organisati-

onsform des Tauschs angestrebt, wobei die Produktionskosten und die Transaktionsatmosphäre als Umwelt der Transaktion als gegeben angenommen werden (vgl. Göbel 2002; Williamson 1985; Ebers, Gotsch 1999). Transaktionskosten werden dabei als sogenannte Reibungskosten bzw. Betriebskosten des Wirtschaftssystems verstanden (Williamson, 1985: 20ff.).

Die Transaktionskostentheorie wurde als Ausgangsbasis gewählt, da in diesem Theoriebereich erstens die Organisationsformen Markt und Hierarchie (Unternehmen)<sup>122</sup>, aber auch von Hybriden/Netzwerken als „kooperative“ Elemente neben Hierarchie (Unternehmen) und Markt besonders untersucht wird (s. Kap. 3; Williamson 1985, 1993). Zweitens operieren alle drei Theoriebereiche des Theoriekonzepts mit Transaktionskosten, so dass die Transaktionskostentheorie als Rahmen dienen kann, um Elemente bzw. Erkenntnisse aus den beiden anderen Theorien zu integrieren. Dies ist besonders für die Betrachtung der Koordinationsform Hierarchie von Bedeutung, da die Analysefähigkeit der Transaktionskostentheorie mittels Elementen der Prinzipal-Agent-Theorie erhöht werden kann (vgl. Göbel 2002; Richter, Furubotn 1996; Erlei et al. 1999).<sup>123</sup>

Die Transaktionskostentheorie wird dem Untersuchungsobjekt Open Source-Projekt entsprechend angepasst, damit eine Analyse der Koordinationsformen bei der Entwicklung umfangreicher und komplexer Software in Zusammenarbeit ermöglicht wird.

#### Vorgehen bei der Interpretation der Theorie bzw. Theoriemodifizierung

Basis der Transaktionskostentheorie ist eine Transaktion, die mit Transaktionskosten, also Aufwand, verbunden ist. Die Koordinationsformen beeinflussen eine Transaktion unterschiedlich. Grundlage der Bestimmung der Ausprägung und des Stellenwerts einer Koordinationsform als Institution ist der Aufwand, den die jeweilige Ausprägung der Koordinationsform hervorbringt. Mittels der Theorie kann aber nicht festgestellt werden, warum gerade bestimmte Ausprägungen als Elemente einer Koordinationsform auftreten. Dazu müssten alle denkbaren Elemente einer Koordinationsform festgestellt und dann auf ihre Effizienz bei der Senkung der Transaktionskosten untersucht werden. Da von dem untersuchten KDE-Projekt ausgegangen wird, sind nicht alle denkbaren Mechanismen bzw. Regelungen wegen hoher Komplexität und begrenzter Rationalität feststellbar. Die auftretenden Koordinationsformen und ihre Elemente sind effiziente Institutionen, die wegen des geringsten Aufwands gegenüber anderen Institutionen überleben konnten. In dieser Theorieinterpretation wird davon ausgegangen, dass ineffiziente Institutionen über Kurz oder Lang verschwinden und durch effizientere ersetzt werden. Die Koordinationsform bleibt somit als Institution bestehen, die am kostengünstigsten ist bzw. den geringsten Aufwand erfordert. Der Stellenwert der Koordinationsform wird dann aus dem Aufwand für alle Elemente der Koordinationsformen bestimmt. Aus der Gegenüberstellung der Koordinationsformen und ihres Aufwands kann dann die Koordinationsform mit der höchsten Bedeutung

---

<sup>122</sup> In der Transaktionskostentheorie wird i.A. von Hierarchie gesprochen, wenn von Unternehmen die Rede ist. In Kap. 3 wird die Begrifflichkeit bürokratische Organisation anstelle von Hierarchie bzw. Unternehmen definiert, wobei Unternehmen eine spezifische Unterorganisationsform ist. Um Begriffsverwirrungen zu vermeiden, wird im Weiteren des Kapitels statt des üblichen Begriffs Hierarchie vor allem der Begriff Unternehmen verwendet. Im Prinzip können aber auch andere bürokratische Organisationen anstelle von Unternehmen diesen Platz in der Neuen Institutionenökonomie einnehmen. Statt des üblicherweise verwendeten Begriffs Hybrid, wird im Weiteren der Begriff Netzwerk verwendet.

<sup>123</sup> Die Prinzipal-Agent-Theorie wird dabei mit der Koordinationsform Hierarchie/Macht in Verbindung gebracht, da der Prinzipal über Eigentum verfügt, von dem der Agent abhängig ist. Der Agent kann zwar aufgrund seines Informationsvorsprungs gegenüber dem Prinzipal Gegenmacht aufbauen, letztendlich hat aber der Prinzipal die größere Machtposition, da dieser den Agenten entlassen kann.

festgestellt werden. Damit können Vermutungen über die Bedeutung einer Koordinationsform durch den Vergleich dieser Koordinationsformen erstellt werden.

Die Vermutung über den Einfluss der Koordinationsformen auf die Transaktion bzw. Transaktionskosten erfolgt in zwei aufeinander folgenden Stufen, die sich jeweils in zwei Zwischenschritten aufteilen. Dabei werden zunächst mittels einer Theoriemodifikation die möglicherweise auftretenden initialen Transaktionsprobleme und der Einfluss der Koordinationsformen vorgestellt (erste Stufe), bevor eine Zusammenführung und Konkretisierung der Erkenntnisse für das Open Source-Projekt KDE erfolgt (zweite Stufe).

In der ersten Stufe (s. Kap. 6.1.1.) wird in einem ersten Zwischenschritt die Transaktionskostentheorie modifiziert und vorgestellt. Die Grundtheorie wird in diesem Zwischenschritt umrissen und stellt den Ausgangspunkt für die weitere Verfeinerung und Anpassung der Theorie an die Koordinationsformen und die Softwareentwicklung in späteren Unterkapiteln dar. Dabei gehen z.T. Konzepte und Ideen der Property-Rights- und Principal-Agent-Theorien in die verwendete Theorie ein. Es wird die Entstehung der initialen Transaktionsprobleme durch eine Kombination von Umwelt- und Verhaltensdeterminanten in einem Unterkapitel (s. Kap. 6.1.1.1.) referiert, wobei diese initialen Transaktionsprobleme in Bereitstellungs-, Such-, Mess- und Spezifitätsprobleme unterteilt werden. Adverse Selection, aber auch Shirking, Free Rider-Verhalten, Hold up, Nachverhandlungen etc. treten dabei auf (s.u.). Dieses allgemeine theoretische Konzept wird in einer Vorüberlegung in einem weiteren Unterkapitel (s. Kap. 6.1.1.2.) verfeinert: Die Art der Bereitstellungs-, Such-, Mess- und Spezifitätsprobleme wird bei hoher Virtualität bzw. ständiger IuK-Technik-Nutzung einer Softwareentwicklung im Vordergrund und dem Open Source-Projekt KDE im Hintergrund vorgestellt.

Im zweiten Zwischenschritt werden in einem Unterkapitel (s. Kap. 6.1.1.3.) Organisationsformen als Beherrschungs- und Überwachungssysteme vorgestellt und erläutert, wie diese in der Theorie normalerweise die initialen Transaktionskosten senken. In dieser Arbeit stehen aber die Koordinationsformen auf einer Handlungsebene im Mittelpunkt, die in diese Theorie eingepasst werden. Daher werden in einem weiteren Unterkapitel (s. Kap. 6.1.1.4.) die Koordinationsformen mit ihren Regelungen betrachtet, die anstelle der Organisationsformen als Beherrschungs- und Überwachungssysteme die initialen Transaktionsprobleme beeinflussen. Der Einfluss jeweils einer einzigen effizienten Koordinationsform steht bei den gleichen initialen Transaktionsproblemen im Mittelpunkt, um diese später vergleichen zu werden (s.u.). Damit kann dann die jeweilige Bedeutung einer Koordinationsform festgestellt werden. Wie später dargestellt wird, kann mit dieser Theorie nicht das Auftreten von spezifischen Elementen der Koordinationsformen bestimmt werden.

In einer zweiten Stufe (s. Kap. 6.1.2.) wird vom hoch virtuellen Open Source-Projekt KDE ausgegangen, in dem die Koordinationsformen vorgefunden wurden. In einem ersten Zwischenschritt werden in einem Unterkapitel (s. Kap. 6.1.2.1.) ausgehend von den Erkenntnissen des ersten Zwischenschritts der ersten Stufe die initialen Transaktionsprobleme im Open Source-Projekt KDE beschrieben. Dabei werden die Bereitstellungs-, Such-, Mess- und Spezifitätsprobleme als initiale Transaktionsprobleme des Open Source-Projekts KDE den Teilprozessen der Softwareentwicklung (Eintritt/Rekrutierung; Arbeitsplanung mit Zieldefinition, detaillierter Entwicklungsplanung, Arbeitsverteilung; Leistungserstellung; Arbeitszusammenführung) zugeordnet. Die gesamte Transaktion besteht dabei aus allen Teilprozessen der Softwareentwicklung.

In einem zweiten Zwischenschritt stehen Vermutungen im Mittelpunkt, die Erklärungsansätze für die auftretenden Koordinationsformen im Open Source-Projekt KDE liefern können. Dabei wird durch einen Vergleich der Koordinationsformen deren jeweiliger Stellenwert analysiert. Verglichen wird, inwiefern die Handlungsbedingung und Handlung einer

Koordinationsform die vorher beschriebenen initialen Transaktionsprobleme in den Teilprozessen der Softwareentwicklung beeinflussen. Für jeden Teilprozess erfolgt dies in einem Unterkapitel. Die Bedeutung der Koordinationsformen kann dann z.B. über die Grenzen der Handlungsbedingung und Handlung einer Koordinationsform bestimmt werden. Außerdem kann die übergreifende Bedeutung einer Koordinationsform für die gesamte Transaktion durch die Summe aller Transaktionskosten über alle Teilprozesse festgestellt werden.

### **6.1.1. Theorieinterpretation der Neuen Institutionenökonomie - Theoretische Grundlagen der Analyse**

In diesem Kapitel 6.1.1. werden jeweils zuerst die Grundlagen dieses Theoriekonzepts vorgestellt und diese dann im Hinblick auf eine Softwareentwicklung bei hoher Virtualität als Schwerpunkt neben dem im Hintergrund stehenden Open Source-Projekt KDE modifiziert. Damit lässt sich diese Theorie ohne große Modifikationen ggf. auch auf eine Softwareentwicklung in anderen virtuellen Arbeitsnetzen oder in Unternehmen anwenden. Die Softwareentwicklung des Open Source-Projekts KDE wird in Kapitel 6.1.2. vertiefend analysiert.

Ausgangspunkt der Theoriemodifikation ist die Transaktionskostentheorie, bei der Transaktionen aus der gesellschaftlichen Tatsache der Arbeitsteilung/Spezialisierung und des Tauschs/der Abstimmung entstehen (vgl. Milgrom, Roberts 1992, 25ff.; Picot et al. 2002: 3ff.). In diesem Zusammenhang wird eine Transaktion als die Überführung eines technisch separierbaren Produkts über eine technisch separierbare Schnittstelle verstanden (Williamson 1985: 1). Der Übertrag bzw. Austausch kann nicht nur physisch in Form einer räumlichen Bewegung von Gütern, sondern auch juristisch in Form der Veränderung von einzelnen Verfügungsrechten erfolgen (vgl. Göbel 2002: 131f.), so dass Transaktionen bei der Zusammenarbeit in einem Unternehmen, aber auch bei Kaufakten auf dem Markt betrachtet werden (können). Bei der Analyse einer Transaktion werden vor allem Unterschiede in den Transaktionskosten ordinal betrachtet und die Produktionskosten als gegeben angenommen.

In einer Transaktion entstehen dabei Transaktionskosten aus zwei Einflüssen. Erstens bestehen initiale Transaktionsprobleme, wie Governance- und Motivationsprobleme, die eine Transaktionsdurchführung erschweren oder verhindern. Die an der Transaktion beteiligten Akteure reagieren zweitens auf diese Gegebenheit durch spezifische Beherrschungs- und Überwachungssysteme, die diese erschwerte Transaktionsdurchführung abmildern sollen, aber wiederum Nutzungskosten hervorrufen. Es geht dabei nicht um das generelle Vermeiden jeglicher Transaktionsdurchführung, sondern um die kostenminimalsten Beherrschungs- und Überwachungssysteme, die durch einen Vergleich festgestellt werden (vgl. bspw. Milgrom, Roberts 1992: 25ff.; Richter, Furubotn 1996: 70).

#### **6.1.1.1. Initiale Transaktionsprobleme als Basiskategorie**

##### Transaktionsdeterminanten und initiale Transaktionsprobleme

Die theoretische Basis stellen Transaktionsdeterminanten dar, die sich aus den Verhaltensannahmen Opportunismus und begrenzte Rationalität auf der einen Seite sowie den Umweltannahmen Unsicherheit/Komplexität und Spezifität/spezifischen Investitionen auf der anderen Seite zusammensetzen. Opportunismus heißt, dass ein Akteur lügen und betrügen, d.h. arglistig handeln kann (Williamson 1985: 47). Begrenzte Rationalität bezieht sich auf die eingeschränkte Aufnahme- und Bearbeitungskapazität des Menschen. Unsicherheit/Komplexität bezeichnet eine Umwelt, die nicht vorbestimmt ist und die wegen vieler zu beachtender Parameter undurchschaubar ist. Spezifität bedeutet, dass spezifische

Investitionen geleistet werden bzw. wurden, die als Sunk Costs eine Rolle spielen, um eine Transaktion durchführen zu können.<sup>124</sup>

In neuerer Literatur wird Vertrauen in die Theorie eingebunden, hat aber bisher keinen Eingang in den Mainstream der Theorie gefunden. In dieser Untersuchung wird einerseits Vertrauen als zentrales Element der Kooperation analog zu dieser Literatur als Gegenstück zu Opportunismus verwendet. Andererseits kann z.B. gegenseitiges Vertrauen Verhaltensunsicherheiten dämpfen bzw. vorhersehbar machen. Dabei besteht eine antiproportionale graduelle Beziehung, wobei Opportunismus und Unsicherheit mit zunehmendem Vertrauen abnehmen und umgekehrt (vgl. Noteboom 2002: 103ff.; Göbel 2002: 123; Williamson 1993; Ripperger 1998).

Kombinationen der genannten Transaktionsdeterminanten lassen initiale Transaktionsprobleme entstehen, die initiale Transaktionskosten verursachen. Initial sind diese Transaktionsprobleme, da die Transaktionsdeterminanten diese Transaktionsprobleme generell hervorrufen, so dass diese schon vor der Durchführung einer Transaktion bestehen. Eine Nulltransaktionskostenwelt ist dabei die Referenzgröße, die ohne den Einfluss der Transaktionsdeterminanten auskommt. Den Bezugspunkt bildet dabei eine Welt ohne transaktionskostenerzeugende initiale Transaktionsprobleme, d.h. eine transaktionskostenlose Welt, wie in der neoklassischen Ökonomie angenommen. In dieser theoretischen Vorstellung wird z.B. immer eine arbeitsteilige Produktion, ein Tausch/Abstimmung sowie eine Motivation der beteiligten Akteure ohne Kosten und damit effizient durchgeführt. Hingegen verringern initiale Transaktionsprobleme durch die damit verbundenen Kosten die Möglichkeit einer (generell effizienten) vertraglichen Transaktionsdurchführung (vgl. Becher et al. 2005: 13f.).

Diese initialen Transaktionsprobleme sind also bei jeder (Kauf-)Entscheidung vorhanden und führen zu initialen Transaktionskosten, die immer zu überwinden sind. Diese initialen Transaktionskosten können aber für verschiedene initiale Transaktionsprobleme unterschiedlich hoch sein. Dabei treten je nach (untersuchter) Transaktion unterschiedliche initiale Transaktionsprobleme mit initialen Transaktionskosten auf. Diese initialen Transaktionsprobleme nehmen in bestimmten Situationen der untersuchten Transaktion unterschiedlich starken Einfluss. Im Allgemeinen wird davon ausgegangen, dass die Summe aller Transaktionskosten aus den initialen Transaktionsproblemen<sup>125</sup> recht hoch ist. Bei einer Kosten-Nutzen-Abwägung eines rationalen Akteurs kann sich dann die Durchführung einer Transaktion nicht für diesen lohnen. Durch die Transaktionsdeterminanten entstehen also Anfangskosten, die es für eine Durchführung einer Transaktion zu überwinden gilt, den Nutzen dieser Transaktion stark übersteigen und diese Durchführung verhindern können. Eine Durchführung einer Transaktion ist nur dann möglich, wenn die initialen Transaktionskosten durch spezifische Instrumente gesenkt werden (s.u.).

### Verschiedene initiale Transaktionsprobleme

Die initialen Transaktionsprobleme bestehen jeweils aus Governance- und Motivationsproblemen (vgl. Abb. 6-1). Unter dem Governanceproblem<sup>126</sup> wird in der transaktionskos-

---

<sup>124</sup> Weitere Annahmen sind der Vollständigkeit halber noch Häufigkeit und Risikoneutralität, die aber in der Transaktionskostentheorie eher peripher behandelt werden und in dieser Arbeit nicht verwendet werden.

<sup>125</sup> Dies kann wie folgt dargestellt werden:  $(\text{Gesamt-TK}_{\text{Initial}} = \text{TK}_{\text{Initial, 1}} + \text{TK}_{\text{Initial, 2}} + \dots + \text{TK}_{\text{Initial, i}}) > 0$

<sup>126</sup> Dieses Problem heißt hier Governanceproblem, um Verwirrungen und Verwechslungen mit der Koordination und den Koordinationsformen zu vermeiden. Hintergrund ist, dass das Governanceproblem in der deutschen transaktionskostentheoretischen Literatur auch Koordinations- oder Abstimmungsproblem

tentheoretischen Literatur ein Bereitstellungs- und ein Suchproblem subsummiert. Das Motivationsproblem kreist um die Erreichung einer Anreizkompatibilität, d.h. möglichst übereinstimmender Motivationen der Akteure bzw. von Prinzipal und Agent, die aber durch ein Messproblem und ein Spezifitätsproblem eingeschränkt werden (vgl. Göbel 2002: 55ff.; Richter, Furubotn 1996; vgl. Becher et al. 2005).

Das Bereitstellungs- und Suchproblem sowie das Mess- und Spezifitätsproblem wird im Weiteren nacheinander aus den Kombinationen von Verhaltens- und Umweltannahmen hergeleitet (Becher et al. 2005: 17ff.; vgl. auch Milgrom, Roberts 1992: 25ff.; Göbel 2002: 55ff., 101ff.; Picot et al. 2002: 5ff.).

- Der Begriff Bereitstellungsproblem bedeutet, dass ein Produkt, eine Information o.ä. überhaupt vorhanden sein und angeboten werden muss. Ein Bereitstellungsproblem besteht, wenn die Transaktionsdeterminanten Spezifität und begrenzte Rationalität in einer Transaktion zusammen auftreten. Ein Beispiel dafür ist, wenn auf dem Arbeitsmarkt keine Bewerber für eine Stelle/Aufgabe vorhanden sind (vgl. Becher et al. 2005).

- Ein Suchproblem bezieht sich auf die Suche eines adäquaten, angebotenen Produkts, einer Information, eines Bewerbers o.ä., wobei diese identifiziert werden müssen. Es hängt mit begrenzter Rationalität der Akteure und Unsicherheit/Komplexität hinsichtlich der Umwelt zusammen und wird durch Spezifität verschärft. Damit ist ein Informationsdefizit über die Beschaffenheit der Umwelt bzw. die Akteure verbunden. Beispielsweise wird auf dem Arbeitsmarkt ein passender Bewerber gesucht, da dieser nicht bekannt ist. Die Eigenschaften eines Akteurs oder Bewerbers bleiben meist verborgen oder werden von diesem in einem vorteilhaften Licht dargestellt. Wenn daher ein „schlechter“ Vertragspartner oder unpassender Bewerber aufgrund von mangelnder Information ausgewählt wird, tritt ein Adverse Selection-Problem auf (Göbel 2002: 101; Arentzen, Lörcher 1997: 61; vgl. Becher et al. 2005).<sup>127</sup>

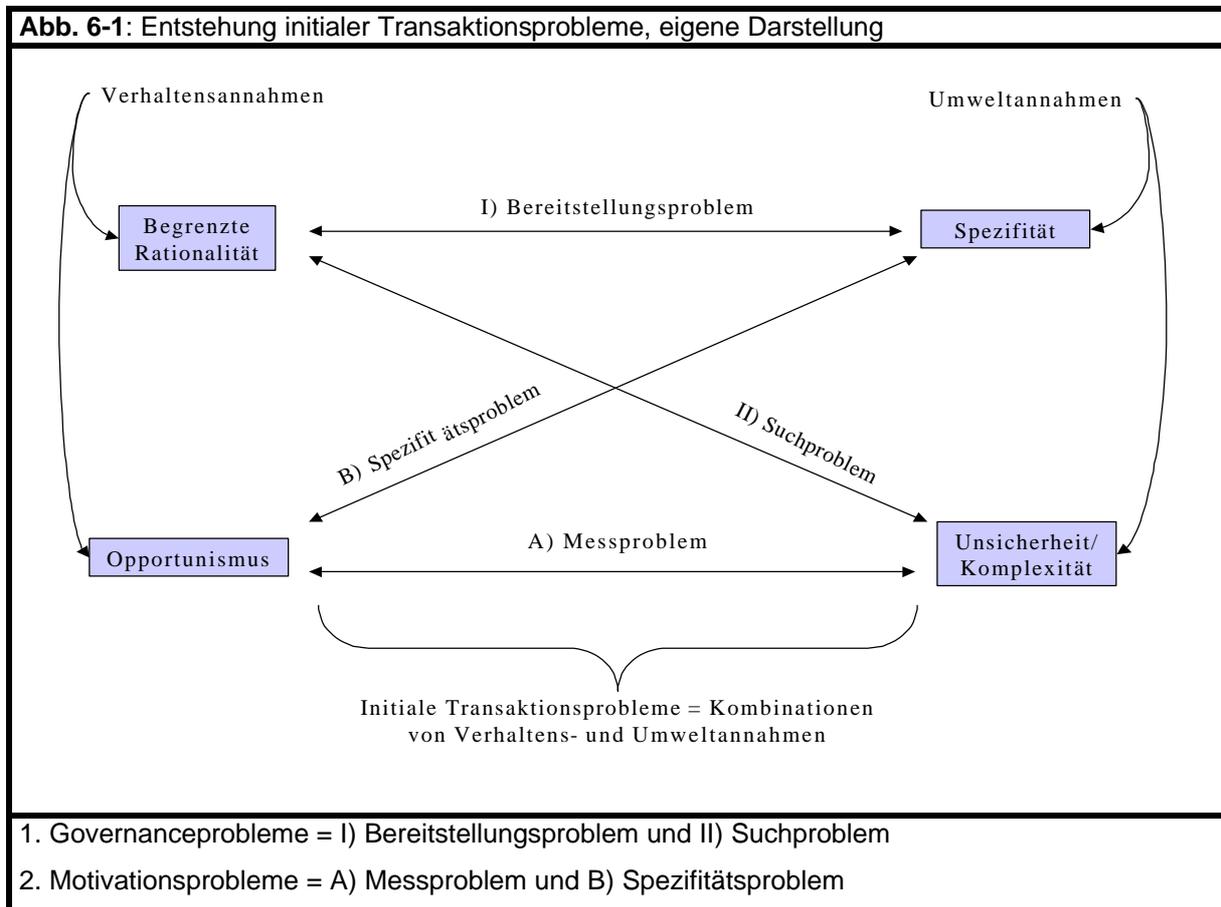
- Ein Messproblem bezieht sich auf die Kontrolle z.B. der Qualität eines Produkts, einer Information, Arbeitsleistung usw. Es entsteht bei Unsicherheit/Komplexität und nicht beobachtbarem bzw. nicht beobachtetem Opportunismus und führt zu einer Informationsasymmetrie. Dies hängt mit einem Moral Hazard<sup>128</sup> zusammen, wobei der besser informierte Partner die Auszahlung (Payoff) des schlechter informierten durch sein Verhalten beeinflusst (Arentzen, Lörcher 1997: 2676). Ein Beispiel ist z.B. das Free-Rider-Verhalten gegenüber gleichgestellten Akteuren, bei dem ein Akteur ein Allgemeingut eigennützig nutzt und dabei die anderen Akteure schädigt (Arentzen, Lörcher 1997:1383; Ebers, Gotsch 1999: 223f.). Weitere Beispiele werden im Rahmen der Prinzipal-Agent-Problematik über u.a. eine suboptimale Ressourcenverwendung von den Autoren diskutiert. Die Investitionsentscheidungen von Managern können als ein Beispiel aus der Literatur durch eigennützige Motive in Bezug auf Fringe Benefits (individueller Zusatznutzen beeinflusst eine Entscheidung, die dadurch suboptimal für die Transaktion ist) wie Machtzuwachs,

genannt wird, wobei die Koordination bzw. Abstimmung in einem Markt stattfindet (vgl. Göbel 2002; Richter, Furubotn 1996).

<sup>127</sup> In der Versicherungswirtschaft wird unter Adverse Selection ein schlechte Wahl der Versicherten (Agenten) vom Versicherer (Prinzipal) gesehen. Nur potenzielle Versicherte/Agenten, die ein hohes Risiko in Bezug zum Versicherungsfall mitbringen, interessieren sich für diese bestimmte Versicherung. Hier wird unter Adverse Selection das generelle Problem der falschen Auswahl durch zu geringe Informationen über den anderen Akteur verstanden.

<sup>128</sup> Dieser Begriff stammt auch aus der Versicherungswirtschaft und bezeichnet das Problem, dass sich die Handlungen des Versicherten (Agenten) nach Vertragsabschluss ändern und einen Versicherungsfall eher heraufbeschwören können (vgl. Arentzen, Lörcher 1997: 2676). Der Begriff entstand in Verbindung mit der Feuerversicherung, wobei der Versicherte nach Vertragsabschluss weniger Sorgfalt bei der Lagerung von Brennbarem walten ließ als vor dem Abschluss. In dieser Arbeit wird der Begriff des Moral Hazard eher in einer breiteren Bedeutung verwendet.

Entscheidung, die dadurch suboptimal für die Transaktion ist) wie Machtzuwachs, Statusgewinn etc. beeinflusst sein, die von den Aktieninhabern nicht festgestellt werden können (s.a. Göbel 2002: 102). Arbeitnehmer als Agenten nutzen als weiteres Beispiel eigennützig das Eigentum des Prinzipals (Consumption-On-The-Job). Arbeitszurückhaltung/Drückebergerei (Shirking) gegenüber Kollegen oder dem Vorgesetzten (Prinzipal) wird auch diskutiert.



- Ein Spezifitätsproblem hängt mit spezifischen Investitionen (sunk costs) z.B. in einer Lieferbeziehung zusammen, womit eine fundamentale Transformation von einem umfangreichen Kreis von Wettbewerbern zu einer Small Numbers Situation bzw. Lock In-Situation<sup>129</sup> zusammenhängt. Durch Opportunismus kann ein Akteur den anderen durch einen Hold Up übervorteilen (vgl. Becher et al. 2005: 17f.; Milgrom, Roberts 1992: 25ff.; Göbel 2002: 55ff., 101ff.; Picot et al. 2002: 5ff.). Ein Spezifitätsproblem besteht auch hinsichtlich informeller bzw. formaler Verträge, da die Vertragspartner ex ante formale, aber auch informale Verträge wegen begrenzter Rationalität und Unsicherheit/Komplexität nur unvollständig abfassen konnten und ex post in ihre Beziehung investiert haben. Durch Umweltveränderungen und ein ein- sowie beidseitiges Informationsdefizit sind Nachverhandlungen bzw. Auslegung auch informaler Elemente vorprogrammiert.

<sup>129</sup> Eine Lock In-Situation tritt ein, wenn durch spezifische Investitionen in eine Beziehung eine Abhängigkeit entsteht. Wenn vorher viele Unternehmen in eine geschäftliche Beziehung eintreten konnten, entwickelt sich durch Investitionen in eine Beziehung eine Small Numbers Situation, d.h. nur wenige mögliche Geschäftspartner sind vorhanden. Durch unterschiedliche Investitionen kann ein Partner den anderen erpressen und ein bestimmtes Verhalten verlangen, was Hold Up genannt wird.

Diese initialen Transaktionsprobleme bestehen vor und nach Vertragsabschluss. Ex ante bestehen Bereitstellungsprobleme und Suchprobleme, die mit Adverse Selection mit Hidden Intention/Hidden Characteristics und der Abfassung unvollständiger Verträge zusammenhängen. Ex post treten Messprobleme in Form von Moral Hazard (Shirking und Free Rider-Verhalten, Fringe Benefits) und Prinzipal-Agent-Problemen sowie Spezifitätsprobleme in Form von Hold Ups, Nachverhandlungen unvollständiger Verträge auf, wobei Hidden Action/Hidden Information eine Rolle spielen.<sup>130</sup> Bei Such- und Messproblemen treten einseitige und beidseitige Informationsdefizite auf. Bereitstellungs- und Spezifitätsprobleme bestehen auch ohne Informationsdefizite, können aber durch diese verschärft werden.<sup>131</sup> Der Ausgleich der Informationsdefizite und Beschaffung zusätzlicher Informationen mildert die Probleme bis zu ihrem Verschwinden.

Auch beim hoch virtuellen Open Source-Projekt KDE können diese initialen Transaktionsprobleme vorkommen. Dabei werden diese Probleme einerseits durch die überwiegend virtuelle Kommunikation verschärft. Andererseits vereinfacht die virtuelle Kommunikation die Informationsbeschaffung für den Ausgleich von Informationsdefiziten oder für zusätzliche Informationen. Wie die IuK-Technik die initialen Transaktionsprobleme verschärft bzw. mildert, wird im folgenden Unterkapitel untersucht.

#### *6.1.1.2. Vorüberlegungen zu initialen Transaktionsproblemen bei einer virtuellen Softwareentwicklung*

In vorangegangenen Kapitel wurde ein grundlegender transaktionskostentheoretischer Abriss hinsichtlich initialer Transaktionsprobleme zu einer kurzen Einführung in diese Thematik dargestellt.

In diesem Unterkapitel wird nun auf die initialen Transaktionsprobleme bei einer virtuellen Softwareentwicklung eingegangen, um diese später bei der Softwareentwicklung im KDE-Projekt identifizieren und einsetzen zu können. Somit wird das Auftreten der oben genannten initialen Transaktionsprobleme bei hoher Virtualität untersucht, d.h. bei einem hohem Nutzungsgrad der IuK-Technik wie dies bei einer Softwareentwicklung der Fall ist. Die Transaktion einer virtuellen Softwareentwicklung steht für eine erste theoretische Konkretisierung im Mittelpunkt, wobei das KDE-Projekt im Hintergrund steht. Es wird kein Vergleich zwischen dem Auftreten initialer Transaktionsprobleme bei IuK-Technik und Nicht-IuK-Technik angestrebt, sondern die ständige Nutzung der IuK-Technik als Basis vorausgesetzt.

Das Auftreten initialer Transaktionsprobleme bei einer hoch virtuellen Softwareentwicklung wird nacheinander anhand des *Bereitstellungs- und Suchproblems* als Governanceprobleme sowie des *Mess- und Spezifitätsproblems* als Motivationsproblem untersucht.

- Auch bei hoher Virtualität bestehen ex ante *Bereitstellungsprobleme* bzgl. Personen/Mitarbeitern, die zwar das Internet nutzen, aber nicht bei der Softwareentwicklung für bestimmte Aufgaben einsetzbar sind. Es nutzen aber immer mehr Personen das Internet,

---

<sup>130</sup> Manche ex post erscheinenden Transaktionsprobleme bestehen nicht oder werden gemildert, wenn das ex ante auftretende Adverse Selection-Problem gelöst wird. Da aber ex post Unsicherheiten auftreten, die eine Verhaltensänderung nach sich ziehen können, treten gerade bei längerfristigen Verträgen bzw. Beziehungen ex post auch andere Transaktionsprobleme auf.

<sup>131</sup> In der Literatur bestehen Uneinigigkeiten bzgl. der Beziehung der Determinanten und einem Hold Up. Manche Autoren sehen diese Situation aus einer Small Numbers-Situation entstehen, die nur mit spezifischen Investitionen und Opportunismus verbunden ist (vgl. z.B. Göbel 2002: 100ff.). Andere dagegen sehen nur Informationsdefizite eine Rolle spielen (vgl. z.B. Picot et al. 2002). Oftmals treten aber Hold Ups ex post auf, wenn Investitionen in eine (Vertrags-)Beziehung durch Umweltänderungen und/oder einseitige Informationsdefizite einen Akteur bevorzugen.

womit im Prinzip immer mehr passende potenzielle Mitarbeiter vorhanden sind und somit das Bereitstellungsproblem über die neuen IuK-Techniken gemildert werden kann (s. z.B. (N)onliner Atlas 2004). Die Nutzung von IuK-Technik ist eine Jedermannsfähigkeit und eine, wenn auch unzureichende Voraussetzung für die Softwareentwicklung. Die komplexe Softwareentwicklung basiert auf spezifischem Wissen/Humankapital, über das u.U. nur wenige Personen verfügen. Die Anzahl der Internetnutzer kann also zunehmen, die Anzahl der Personen mit entsprechendem Wissen/Humankapital aber z.B. auf sehr niedrigem Niveau stagnieren.

Ein weiteres Bereitstellungsproblem besteht, wenn die potenziellen Mitarbeiter nichts von der Softwareentwicklung wissen. Dies kann auf den mangelnden Zugriff auf schon digitalisierte Daten<sup>132</sup> zurückgeführt werden, was aber durch die Standardisierung im Internet kaum eine Rolle spielt. Ein weiteres Problem besteht, wenn Daten oder Informationen nicht in digitaler Form vorhanden sind. Einerseits werden digitalisierbare Daten oder Informationen wegen fehlender Motivation nicht digitalisiert und kommuniziert. Andererseits können Daten oder Informationen wie Soft Skills bei Bewerbungen nicht oder nur schwer digitalisiert werden (s. Becher et al. 2005: 20). Dies ist besonders bei der Softwareentwicklung der Fall, wobei das Wissen/Humankapital einer Person einen hohen Stellenwert hat und sich schwerlich digitalisieren lässt. Außerdem besteht gerade bei der virtuellen Kommunikation z.B. bei einer Softwareentwicklung ein Informationsparadoxon<sup>133</sup>, welches mit dem Informationsgut-Charakter der Software zusammenhängt (vgl. Shapiro, Varian 1998). Dies erschwert eine Bewertung der Information oder Software ex ante, ist u.a. für eine mangelnde Bereitstellung verantwortlich und erhöht die Transaktionskosten stark.

- *Suchprobleme* werden ex ante einerseits durch die IuK-Techniken gemildert, andererseits verstärkt. Die Technik unterstützt z.B. die Qualität und Schnelligkeit der Informationsübertragung, Auswahl und Bewertung der Daten durch standardisierte Filter und Suchmaschinen. Dies ermöglicht eine bessere Information über Transaktionspartner. Außerdem können durch die verringerten räumlichen Schranken vermehrt Informationen über Alternativen bzw. Transaktionspartner erreicht werden. Damit wird die Suche stark begünstigt, da die ein- und beidseitige Informationsunsicherheit gesenkt wird und somit die Adverse Selection-Problematik seltener auftritt (vgl. Becher et al. 2005: 19ff.; Ciborra, Olson 1988). Weiterhin kann die IuK-technische Infrastruktur die Kommunikation durch Fragebögen, Filter und die Softwareproduktion über eine Modularität der Software standardisieren. Auch manche nachfolgenden Motivationsprobleme werden gemildert, da vermehrt Informationen über das vorherige Verhalten und dadurch über die möglichen Motivationen der Akteure vorhanden sind, solange die Digitalisierbarkeit der Daten gegeben ist. Mitarbeiter können mit einer spezifischen Kombination von Anreizen bzw. Motivationspräferenzen ausgewählt werden, die von monetärem Einkommen und dessen Höhe (extrinsische Motivation) bis zu nicht-monetären Elementen wie Bastelmentalität (intrinsische Motivation) reichen (vgl. Serries 2005).

Die vereinfachte und kostengünstige Suche führt aber wiederum dazu, dass mehr Informationen, darunter auch irrelevante, eingeholt werden können. Initiale Transaktionsprobleme

---

<sup>132</sup> Der stark erschwerte Zugriff auf digitalisierte Daten hängt z.B. mit inkompatiblen Datenformaten/Software, Inselcomputern ohne Anschluss an das Internet oder anderen IuK-technischen Problemen zusammen und generiert zusätzliche Kosten.

<sup>133</sup> Ein Informationsparadoxon ist ein Bewertungsproblem, bei dem der Wert einer Information erst festgestellt werden kann, wenn die Information bekannt ist. Ist sie bekannt, besteht kein Anreiz mehr, für diese eine Gegenleistung zu entrichten (Arentzen, Lörcher 1997: 1878). Dabei spielt eine asymmetrische Informationsverteilung zwischen Käufer und Verkäufer, verbunden mit einem Spezifitätsproblem eine Rolle, da durch den Kauf Sunk Costs entstehen, der Inhalt der Information aber vollkommen nutzlos sein kann.

der Auswahl werden verstärkt, da durch die räumliche Schrankenlosigkeit der IuK-Technik mit einer hohen Anzahl an Informationen bzw. potenziellen Transaktionspartnern ein Information Overflow ermöglicht wird. Die hohe Anzahl an Informationen kann wegen der begrenzten Rationalität der Akteure nicht bearbeitet werden. Es bestehen kaum Möglichkeiten, relevante von irrelevanten bzw. falschen Informationen zu unterscheiden. Dies führt direkt zum weiteren Problem der Anonymität/Glaubwürdigkeit bei standortungebundener Kommunikation, da durch die Virtualität keine Überprüfbarkeit der Person möglich ist und Akteure Transaktionsgewinne abschöpfen können, ohne Gegenleistungen zu erbringen oder Sanktionen fürchten zu müssen. Die Adverse Selection-Problematik<sup>134</sup> wird wie oben angenommen nicht aufgehoben, sondern verstärkt (vgl. Becher et al. 2005: 20, 27f.; Kumbruck 2000).<sup>135</sup>

Folglich kann das Auftreten von Information Overflows verringert, aber nicht verhindert werden. Auf Seiten der Mitarbeiter besteht daher das individuelle Kalkül, dass einerseits die neuesten Aktivitäten zu beobachten sind, andererseits die unwichtigen Informationen herausgefiltert werden müssen, um nicht von der Produktion abgehalten zu werden.

- Ex post ergibt sich ein *Messproblem* mit dem nachfolgenden Problem der Vergleichbarkeit bei der Zusammenarbeit in einer Softwareentwicklung. Moral Hazard oder die Prinzipal-Agent-Problematik hängen mit einer Informationsasymmetrie zusammen, so dass ein Vertragspartner opportunistisch handeln kann. Eine Informationsasymmetrie besteht auf der einen Seite bzgl. des Arbeitseinsatzes der Arbeitskraft (Shirking) und auf der anderen hinsichtlich der Ressourcen des KDE-Projekts wie Free Rider-Verhalten, Fringe Benefits und Consumption-On-The-Job. Die Shirking-Problematik besteht gerade in der Softwareentwicklung hoch komplexer Erfahrungsgüter, da Arbeitsfähigkeit (Umfang und Qualität der Umsetzung potenzieller in tatsächliche Leistung), Arbeitseinsatz und Wissen/Humankapital einer Person schwer zu digitalisieren sind. Free Rider-Verhalten bezieht sich auf die gemeinsame, prinzipiell gleichgestellte Nutzung eines öffentlichen Guts, wobei ein Akteur seine wirklichen Präferenzen verdeckt, um nicht für die Finanzierung/Aufrechterhaltung des Guts herangezogen zu werden. Er erhöht den eigenen Nutzen auf Kosten der anderen (Arentzen, Lörcher 1997: 1383). Das Free Rider-Verhalten ist bei der Softwareentwicklung abhängig von der ansteigenden Anzahl der Mitarbeiter bei gleichzeitiger Anonymität schwieriger nachzuweisen, da die gleichzeitige Beobachtung aller Mitarbeiter wegen der begrenzten Rationalität kaum möglich ist. Bei Fringe Benefits beeinflussen individuelle Zusatznutzen eine Entscheidung, die damit suboptimal für die Transaktion ist.<sup>136</sup> Diese Begrifflichkeit ähnelt stark dem Free Rider-Verhalten und wird mit diesem im Weiteren diskutiert. Hintergrund ist, dass die Open Source-Mitarbeiter aufgrund einer Informationsasymmetrie ihren eigenen Nutzen verfolgen können. Dabei kann eine Entscheidung, die suboptimal für das untersuchte Open Source-Projekt ist, als eine Schädigung im Sinne eines Free Rider-Verhaltens verstanden werden. Da das untersuchte Open Source-Projekt außerdem kein Unternehmen ist, aber ein (quasi-)öffentliches Gut bei prinzipiell Gleichgestellten produziert, steht das Free Rider-Verhalten begrifflich eher im Mittelpunkt. In Unternehmen wird auch von Consumption-On-The-Job gesprochen, wenn

---

<sup>134</sup> Daraus kann ein Zusammenbruch des Markts erfolgen, da sich die Merkmale nicht feststellen lassen (vgl. Market-for-Lemons-Problematik; s.a. Arentzen, Lörcher 1997: 61; Akerloff 1970).

<sup>135</sup> Eine Person kann z.B. unter falschem Namen bei Ebay ein Gut anbieten, das Geld des Käufers „kassieren“ und dann „verschwinden“.

<sup>136</sup> Fringe Benefits als individueller Zusatznutzen bezieht sich in der Literatur üblicherweise auf eine Vertragsbeziehung zwischen wirtschaftlichen Akteuren auch in einem Unternehmen, wobei ein Agent eine Alternative für seine eigene Nutzenerhöhung wählt, für den Prinzipal jedoch eine andere Alternative optimaler wäre, die er aber nicht feststellen kann. Dies wird in dieser Arbeit etwas allgemeiner betrachtet.

ein Agent die Ressourcen des Prinzipals zu seinem eigenen Nutzen missbraucht (vgl. Göbel 2002: 101ff.). In dieser Arbeit ist aber Consumption-On-The-Job nicht relevant, da dies in einem Arbeitsnetz oder Open Source-Projekt nicht auftritt, weil die einzelnen Akteure rechtlich eigenständig sind und über Internet keinen Zugriff auf die Ressourcen des Prinzipals haben.

- Ein Beispiel für *Spezifitätsprobleme*<sup>137</sup> ist eine sinkende Zeitspezifität bei durch die IuK-Technik beschleunigten Transaktionen. Zeitspezifität heißt, dass Investitionen für einen bestimmten Termin getätigt werden, die danach relativ wertlos sind (Ebers, Gotsch 1999: 228; Göbel 2002: 140). Durch die IuK-Technik lassen sich Termine bei der Softwareentwicklung, zu denen z.B. ein bestimmtes Gut abgegeben werden soll, in kürzeren Zeitabständen anberaumen. Kontrollen können zeitnäher erfolgen. Dadurch können aber auch häufiger Hold Ups bei der Softwareentwicklung auftreten, da zu bestimmten Terminen Software fertig werden soll. Da die Termine durch z.B. vertragliche Regelungen einzuhalten sind, kann z.B. ein Vertragspartner den anderen erpressen, bestimmte ansatzweise programmierte Funktionalitäten der Software für den Termin auszuklammern bzw. überhaupt nicht weiter zu verfolgen oder bestimmte Funktionalitäten weiterzuverfolgen. Mit der sinkenden Zeitspezifität kann sich zugleich die Wahrscheinlichkeit für das Auftreten von Information Overflows erhöhen. Begrenzte Rationalität verhindert dabei bei der ständigen Beachtung der Termine eine effiziente Informationsauswahl.

Außerdem bauen die Mitarbeiter projektspezifisches Humankapital auf (Humankapitalspezifität), was die Wahrscheinlichkeit eines Hold Ups erhöht. Dadurch sind ex post nicht nur Hold Ups von den Mitarbeitern, sondern auch vom Prinzipal möglich (u.U. Machtzuwachs), was zu einer wechselseitigen Abhängigkeit u.a. verbunden mit einer fundamentalen Transformation führen kann.

Hold Ups können auch aus Investitionen in die Reputation des KDE-Projekts erfolgen, wobei der Name und Logos dieses Projekts als Marken verstanden werden können (Markennamenspezifität). Dabei kann ein Mitarbeiter durch Androhung von Fehlverhalten diese Investition gefährden und seinen Willen durchsetzen (vgl. Ebers, Gotsch 1999: 228; Göbel 2002: 140f.).

Die IuK-technische, räumliche Schrankenlosigkeit ermöglicht eine geringere Standortspezifität (vgl. Göbel 2002: 139f.; Williamson 1991: 281), die zu einer Erhöhung der Anzahl der potenziellen Transaktionspartner und Mitarbeiter führt und damit die Bereitstellungproblematik abmildert. Dadurch treffen sich viele Gleichgesinnte und bilden virtuelle Gruppenforen oder virtuelle Arbeitsnetze wie das KDE-Projekt. Umgekehrt kann die Bereitstellungproblematik auch auf das Spezifitätsproblem wirken: Ein Hold Up kann von Mitarbeitern mit spezifischem Humankapital ausgehen, wenn kaum potenzielle Mitarbeiter mit spezifischem Humankapital z.B. für spezifische Softwareentwicklung auf dem Arbeitsmarkt oder sonst wie vorhanden sind (Monopolstellung).

Ein weiteres Spezifitätsproblem besteht, wenn Vertragspartner in ihre Beziehung investiert haben, ex ante jedoch keinen vollständigen Vertrag abschließen konnten. Zusätzliche Kosten entstehen durch die Nachverhandlungen informaler und formaler Elemente dieses unvollständigen Vertrags, wenn z.B. Umweltveränderungen auftreten. Auftretende Machtverschiebungen zwischen den Akteuren können dann zu Nachverhandlungen und weiteren Hold Ups führen (vgl. Göbel 2002: 140f.). In Zusammenhang mit weiteren Spezifi-

---

<sup>137</sup> Spezifitätsprobleme beziehen sich auf Abhängigkeiten, die durch eine zu hohe Investition in die Beziehung bestehen. Hold Ups können als schädigende Machtverwendung bzw. in einigen Fällen „Machtmissbrauch“ bezeichnet werden, beeinträchtigen eine Transaktionsdurchführung und führen zu einer suboptimalen Kostenverteilung.

tätsproblemen werden auch anlage- und abnehmerspezifische Investitionen genannt (vgl. Ebers, Gotsch 1999: 228). Die autonomen Arbeitsnetzmitarbeiter investieren, wenn sie wiederholt temporär zusammenarbeiten, in spezielle Infrastruktursoftware oder halten Arbeitskapazitäten frei. Dies kann von einem der Mitarbeiter oder vom Prinzipal für einen Hold Up verwendet werden.

Zusammengefasst lässt sich sagen, dass die IuK-Technik z.B. durch erhöhten Informationszugriff bzw. räumliche Schrankenlosigkeit z.T. initiale Transaktionsprobleme bis zu einem gewissen Grad abmildert, jedoch auch neue schafft. Die einzelnen Problematiken stehen z.T. in Wechselwirkung miteinander. Die ex ante initialen Transaktionsprobleme beeinflussen die ex post Probleme, z.B. das Bereitstellungs- das Spezifitätsproblem (s.o.). Weitere technisch-vermittelte oder informale/formale Regeln wie die in Kapitel 3 beschriebenen Koordinationsformen sind nötig, um diese Probleme abzumildern oder zu vermeiden. Im Folgenden wird daher gezeigt, wie die Koordinationsformen auf die initialen Transaktionsprobleme wirken und die damit verbundenen Transaktionskosten senken.

#### *6.1.1.3. Senkung von Transaktionskosten aus initialen Transaktionsproblemen durch Beherrschungs- und Überwachungssysteme*

Transaktionsdeterminanten lassen also initiale Transaktionsprobleme entstehen, die Transaktionskosten nach sich ziehen, die eine Produktion bzw. einen Austausch von Gütern verhindern, verzögern oder beeinträchtigen. Die Durchführung einer Transaktion wird dadurch verteuert und u.U. verhindert. Bei Überwindung der initialen Transaktionsprobleme bleibt somit der Nutzen einer Transaktion oftmals weit hinter dem aus der Transaktion entstehenden Schaden und den Kosten zurück. Übertragen auf den Kostenaspekt der Transaktion besteht ein negativer Gewinn aus niedrigem Erlös und hohen Transaktionskosten. Die Transaktion wird dabei entweder gar nicht durchgeführt oder es wird bei Unsicherheiten ex post festgestellt, dass die Transaktion mit negativem Gewinn erfolgte. Da aber nur Institutionen bestehen bleiben, aus denen ein positiver Gewinn entsteht und die effizienter als andere Institutionen sind, d.h. einen höheren Gewinn als andere haben, wird diese Überlegung nicht weiter verfolgt.

Die initialen Transaktionskosten, die durch die initialen Transaktionsprobleme entstehen, können also den Nutzen einer Transaktion bei weitem übersteigen, so dass eine Durchführung verhindert wird. Die Akteure etablieren deswegen Beherrschungs- und Überwachungssysteme in Form von Institutionengeflechten bzw. verschiedenen „Organisationsformen“<sup>138</sup>, die den Nutzen erhöhen bzw. die Transaktionskosten verringern. Die initialen Transaktionsprobleme werden dadurch überwunden und es etabliert sich eine neue Situation, bei der zwar die verringerten Transaktionskosten aus den Transaktionsdeterminanten noch bestehen, aber die Kosten kein unüberwindliches Hindernis mehr darstellen (vgl. Becher et al. 2005).

Die Beherrschungs- und Überwachungssysteme senken über spezifisch ausgerichtete Governance- und Motivationsinstrumente<sup>139</sup> die Transaktionskosten aus Governance- und Motivationsproblemen. Ein Bereitstellungsproblem, z.B. dass keine genau passenden Arbeitnehmer für eine Stelle/Aufgabe vorhanden sind, kann durch das Instrument Fortbildung bzw. Nachqualifizierung oder durch das Abwälzen auf das Suchproblem mit der Su-

---

<sup>138</sup> Unter Organisationsformen werden in der Transaktionskostentheorie Markt und Unternehmen sowie Hybridformen verstanden (Göbel 2002: 143ff.).

<sup>139</sup> Alle Einflüsse heißen Instrumente, die diese Transaktionskosten senken. Diese Einflüsse können einerseits instrumentelle Eingriffe, aber auch soziale, ungezielt entstandene Gegebenheiten sein.

che nach Personen mit ähnlichem Humankapital gelöst werden. Das Suchproblem kann z.B. durch Signalling (der Signalisierung bestimmter Merkmale bzw. Qualifikationen durch die Bewerber), Self Selection (der Bewerbung nur passender Personen auf ein Angebot des Arbeitgebers) und Screening (der aktiven Informationserhebung des Arbeitgebers über den Bewerber) gemildert werden. Ein Messproblem kann durch Monitoring (d.h. Verhaltenskontrolle) oder durch Reporting (ein Akteur meldet und dokumentiert seinen Aufwand, damit sein Erfolg von Misserfolgen durch exogene Einflüsse unterschieden und bewertet wird) entschärft werden (Göbel 2002: 110ff.). Transaktionskosten aus den initialen Transaktionsproblemen können auch durch Vertrauen<sup>140</sup> gesenkt werden, das Opportunismus entgegenwirkt und zusätzlich das Auftreten von Transaktionsproblemen verhindert, bei denen Unsicherheit eine Rolle spielt (Noteboom 2002: 103ff.; Göbel 2002: 123). Die möglichen Kostenverluste aus einem Spezifitätsproblem werden durch die Suche/Wahl eines anderen Vertragspartners gemäßigt.

Die Nutzung dieser Abstimmungs- und Motivationsinstrumente bei der Durchführung einer Transaktion verursachen wiederum Transaktionskosten (sog. Instrumentnutzungskosten), so dass die restlichen, nicht abbaubaren Transaktionskosten der initialen Transaktionsprobleme mit den Nutzungskosten der Instrumente die Gesamttransaktionskosten bilden (vgl. Abb. 6-2). Die Gesamttransaktionskosten sind dabei alle Kosten, die hinsichtlich einer Transaktion (z.B. eine Softwareproduktion) entstehen. Diese Instrumente senken aber die Gesamttransaktionskosten nur bis zu einem gewissen Grad, da die Nutzung der Instrumente selbst wieder Transaktionskosten verursacht.<sup>141</sup>

Damit ein effizientes Instrument verwendet wird, muss aber die Transaktionskostensenkung der initialen Transaktionsprobleme größer als die Entstehung von Transaktionskosten durch die Instrumentnutzung sein.

Die Gesamttransaktionskosten werden also nicht nur durch die Höhe der initialen Transaktionskosten, sondern auch durch die Beherrschungs- und Überwachungssysteme beeinflusst. Eine Organisationsform als Beherrschungs- und Überwachungssystem beinhaltet spezifische Instrumente, die sich je nach Organisationsform bzw. Beherrschungs- und Überwachungssystem unterscheiden und daher unterschiedlich auf die initialen Transaktionsprobleme wirken. Die Beherrschungs- und Überwachungssysteme beinhalten dabei auch Koordinationsformen, die wiederum aus unterschiedlichen Mechanismen bzw. Regelungen bestehen.

In dieser Arbeit liegt der Untersuchungsfokus nicht auf der Organisationsform als Beherrschungs- und Überwachungssystem auf der Mesoebene, sondern auf den Koordinationsformen Kooperation, Hierarchie/Macht und Wettbewerb mit ihren spezifischen unterschiedlichen Mechanismen bzw. Regelungen auf der Handlungsebene (s. Kap. 3).

---

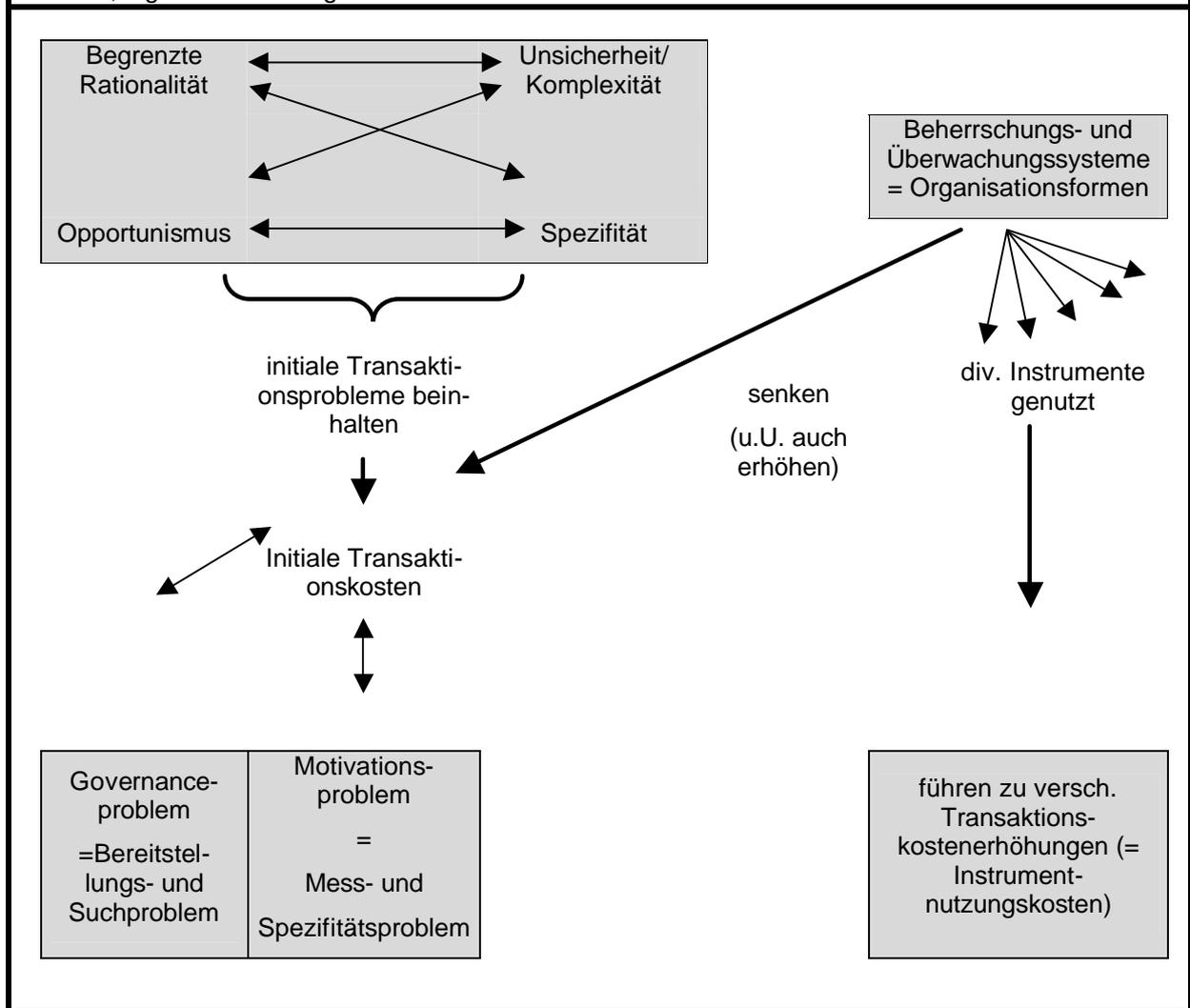
<sup>140</sup> Vertrauen ist eine soziale Gegebenheit, die Transaktionskosten senkt, aber schwerlich instrumentell zu nutzen ist.

<sup>141</sup> Diese Zusammenhänge werden in folgender Gleichung deutlich:

$$\begin{aligned} TK_{\text{Gesamt}} &= (\text{anfängliche}) TK_{\text{Initial}} + N_{\text{Instrument/Beherrsch.system, } i} + TK_{\text{Instrument/Beherrsch.system, } i} \\ &= TK_{\text{Initial}} - TK_{\text{Instrument/Beherrsch.system, } i} + TK_{\text{Instrument/Beherrsch.system, } i}; \text{ dabei:} \\ &N_{\text{Instrument/Beherrsch.system, } i} \text{ entspricht } - TK_{\text{Instrument/Beherrsch.system, } i} \end{aligned}$$

TK heißt Transaktionskosten, N heißt Nutzen,  $TK_{\text{Initial}}$  sind die Initialen TK,  $TK_{\text{Instrument/Beherrsch.system}}$  sind die TK des Beherrschungs-/Überwachungssystems;  $i$  = Markt, Unternehmen, Hybrid

**Abb. 6-2:** Initiale Transaktionsprobleme und Beherrschungs/Überwachungssysteme; nach Becher et al. 2005, eigene Darstellung



In der bisherigen Literatur über die Transaktionskostentheorie wurde das Beherrschungs- und Überwachungssystem in Form der Organisationsformen Markt, Unternehmen (Hierarchie) und Hybrid mit der höchsten Effizienz für eine Transaktion festgestellt, indem die Gesamttransaktionskosten der jeweiligen Organisationsform zusammengerechnet und miteinander verglichen wurden (s.a. Make-or-Buy-Entscheidung). Dabei wurde, wie schon in Kapitel 3 argumentiert, die Organisationsformen mit bestimmten Instrumenten in Verbindung gebracht, d.h. Markt mit Wettbewerb, Unternehmen mit Hierarchie und Hybridform mit Kooperation gleichgesetzt. Die transaktionskostenminimalste Organisationsform wurde dann für die Transaktion gewählt bzw. setzte sich als effizienteste Organisationsform für die Transaktion durch (vgl. Göbel 2002: 143ff.; Williamson 1985, 1991; Richter, Furubotn 1996).

In dieser Untersuchung wird an den Koordinationsformen und nicht den Organisationsformen angesetzt, was mit relativ wenigen Veränderungen an den Grundlagen der Theorie möglich ist. Der Einfluss verschiedener Koordinationsformen mit ihren Mechanismen bzw. Regelungen auf die initialen Transaktionsprobleme wird daher im Weiteren analysiert, da

in dieser Arbeit generell die Koordination in einer spezifischen Organisationsform, dem KDE-Projekt, im Mittelpunkt steht.<sup>142</sup>

#### *6.1.1.4. Einfluss von Koordinationsformen auf die initialen Transaktionsprobleme bei einer virtuellen Softwareentwicklung*

Nachdem die Beherrschungs- und Überwachungssysteme als grundlegende Elemente im Theoriekonzept dargestellt wurden, wird diese Theorie im Folgenden modifiziert und die Koordinationsformen statt Organisationsformen eingepasst. Dieses Vorgehen orientiert sich an dem Vorgehen in der Transaktionskostentheorie. In der Transaktionskostentheorie wird untersucht, welche Form der Organisation (Markt, Unternehmen, Hybrid/Netzwerk) als Beherrschungs- und Überwachungssystem bei welcher Art von Transaktion angemessen ist. Dies lässt sich auch auf die Koordination übertragen, wobei die Koordinationsformen in der Theorie als Instrumente anstelle der jeweiligen Beherrschungs- und Überwachungssysteme betrachtet werden. Das jeweilige Beherrschungs- und Überwachungssystem stellt dann die übergeordnete institutionelle Ebene dar, in deren Einflussbereich die Koordinationsformen als untergeordnete Institutionen wirken und die initialen Transaktionsprobleme beeinflussen (vgl. Göbel 2002: 139ff.). Die Koordinationsformen beeinflussen also wie die Beherrschungs- und Überwachungssysteme die Gesamttransaktionskosten, indem sie die initialen Transaktionskosten senken und dadurch die Überwindung der initialen Transaktionsprobleme ermöglichen.

Ausgangspunkt ist, dass die Koordinationsformen die initialen Transaktionskosten senken, die aus den initialen Transaktionsproblemen entstehen, aber zugleich durch ihre Benutzung Transaktionskosten erhöhen. Der Einsatz einer Koordinationsform als Beherrschungsinstrument mildert bzw. überwindet somit diese initialen Transaktionsprobleme, wobei diese Milderung als der Nutzen dieser Koordinationsform betrachtet werden kann.

Die Koordinationsformen haben u.a. einen mittelbaren Einfluss auf die Produktionskosten bzw. die Produktqualität<sup>143</sup>, der aber nicht weiter betrachtet wird. Die Produktionskosten bzw. die Produktqualität werden also im Weiteren bei allen drei Koordinationsformen als gleich hoch angesehen, da bei der Softwareentwicklung im Zusammenhang mit den drei Koordinationsformen die gleiche Software erstellt wird.

---

<sup>142</sup> Wurde bei der Feststellung der Koordinationsformen über eine hohe Institutionaliserungsstärke in Form von langer Dauer, hoher Personenanzahl und hoher Frequenz argumentiert, gehen diese Indikatoren auch in dieses Theoriekonzept ein. Doch wird in der Theorie anders herum argumentiert. Eine lange Dauer, hohe Personenanzahl und hohe Frequenz führen zu hohen Transaktionskosten, die durch die Senkung der initialen Transaktionskosten ausgeglichen werden müssen.

<sup>143</sup> Dieser mittelbare Einfluss wird in dieser Untersuchung ausgeklammert, weil die Höhe des Einflusses durch die Koordinationsformen kaum festzustellen ist: Da die Produktion normalerweise nicht unabhängig von der Transaktion ist, verbessern die Koordinationsformen normalerweise die Produktqualität und erhöhen somit die Produktionskosten unterschiedlich. Da aber die Produktqualität bzw. Produktionskosten konstant sind, sind die Transaktionskosten bei den Koordinationsformen unterschiedlich niedrig bzw. werden unterschiedlich gesenkt, d.h. diejenige Transaktion ist bei einer Koordinationsform effizienter, die bei gleichen Produktionskosten die geringsten Transaktionskosten generiert. Oder umgekehrt: Es werden unterschiedlich hohe Transaktionskosten bei den Koordinationsformen benötigt, um die gleichen Produktionskosten zu erreichen. Außerdem wird in dieser Arbeit die Softwareentwicklung im Arbeitsnetz Open Source-Projekt untersucht, wobei ein Softwareprodukt entsteht. Es ist dabei kaum festzustellen, wie die Produktionskosten bei Verwendung nur einer Koordinationsform sind.

Genauere Betrachtung und vorausgehende Bemerkungen zum Einfluss der Handlungsbedingung und Handlung von Koordinationsformen

In diesem Abschnitt steht im Mittelpunkt, wie die Koordinationsformen die Transaktion bzw. initialen Transaktionsprobleme theoretisch beeinflussen, d.h. wie diese die Transaktionskosten senken oder erhöhen können. Diese Darstellung dient dann als Basis für die spätere, konkrete Analyse der Koordination in einem Open Source-Projekt.

Für jede Koordinationsform wird zuerst in einem eigenen, ersten Abschnitt eine übergreifende Perspektive eingenommen, wobei von der konkreten Transaktion in einer Organisationsform weitestgehend abstrahiert wird. Im Mittelpunkt des Interesses steht die jeweilige Koordinationsform mit der möglichen Wirkungsweise der charakteristischen Koordinationsformenelemente, unterteilt nach Handlungsbedingung und Handlung. Es wird nacheinander für jede Koordinationsform diskutiert, inwiefern die Handlungsbedingung und Handlung einer Koordinationsform überhaupt initiale Transaktionsprobleme mildern können und zugleich durch ihre Nutzung auch Instrumentnutzungskosten verursachen.<sup>144</sup> Ein möglicher Einfluss der Elemente einer Koordinationsform wird vorgestellt und nicht der dezidierte Einfluss auf die einzelnen Governance- und Motivationsprobleme.

Die Transaktionsveränderungen sind zusätzlich davon abhängig, welche Akteure an einer Transaktion beteiligt sind bzw. zu dieser gerechnet werden. Die Kosten von möglichem schädigendem Verhalten von Akteuren gegenüber transaktionstragenden Akteuren gehen u.U. in die initialen Transaktionskosten ein und erhöhen diese. Transaktionstragende Akteure sind dabei die Akteure, die zu einer Transaktion gehören und diese voranbringen. Nicht-transaktionstragende bzw. transaktionsexterne Akteure unterstützen die Transaktion nicht direkt.

Die Handlungsbedingungen der Koordinationsformen, die vor allem die initialen Transaktionskosten senken, sind bei Kooperation das Vertrauen, bei Hierarchie/Macht ein Über-/Unterordnungsverhältnis und bei Wettbewerb die (gegenseitige) Beobachtung (s. Abb. 6-3). Dabei bestehen z.B. Einflüsse auf die unvollständige Information eines Akteurs oder mehrerer Akteure. Andere Mechanismen bzw. Regeln, die auch die initialen Transaktionskosten z.B. in Form von Informationsdefiziten verringern, aber auch Instrumentnutzungskosten generieren, sind bei Kooperation die Verhandlung mit Konsens, bei Hierarchie/Macht Kontrolle und Machteinsatz und bei Wettbewerb der Vergleich.

Zusätzlich wird für Handlungsbedingung und Handlung festgestellt, unter welchen Voraussetzungen diese die Transaktionskosten senken, aber auch erhöhen können. Auf diese Transaktionskostenänderungen wirkt das Handlungsergebnis einer Koordinationsform ein. Das Handlungsergebnis kann zwar nicht empirisch festgestellt werden (vgl. Kap. 3.3.), wirkt sich aber indirekt auf die Transaktion aus. Akteure streben je nach Koordinationsform unterschiedliche Handlungsergebnisse in einer Transaktion an, wobei je nach Koordinationsform unterschiedlich viele Akteure eine Nutzenerhöhung erfahren. Die Akteure, deren Nutzen erhöht wird, gehören zur Transaktion, die Akteure, deren Nutzen sinkt, zählen nicht dazu. Umgekehrt betrachtet, haben die Akteure, die zur Transaktion, also der Softwareentwicklung im KDE-Projekt, gehören eine Nutzenerhöhung erhalten. Der Gesamtnutzen aller Akteure, deren Nutzen erhöht wird, kann dabei als eine Transaktionskostensenkung in einer Transaktion eingestuft werden.

Bei Kooperation ist die Nutzenerhöhung durch eine Win-Win-Situation gekennzeichnet, wobei alle Beteiligten von einer Nutzenerhöhung bzw. Senkung initialer Transaktionskos-

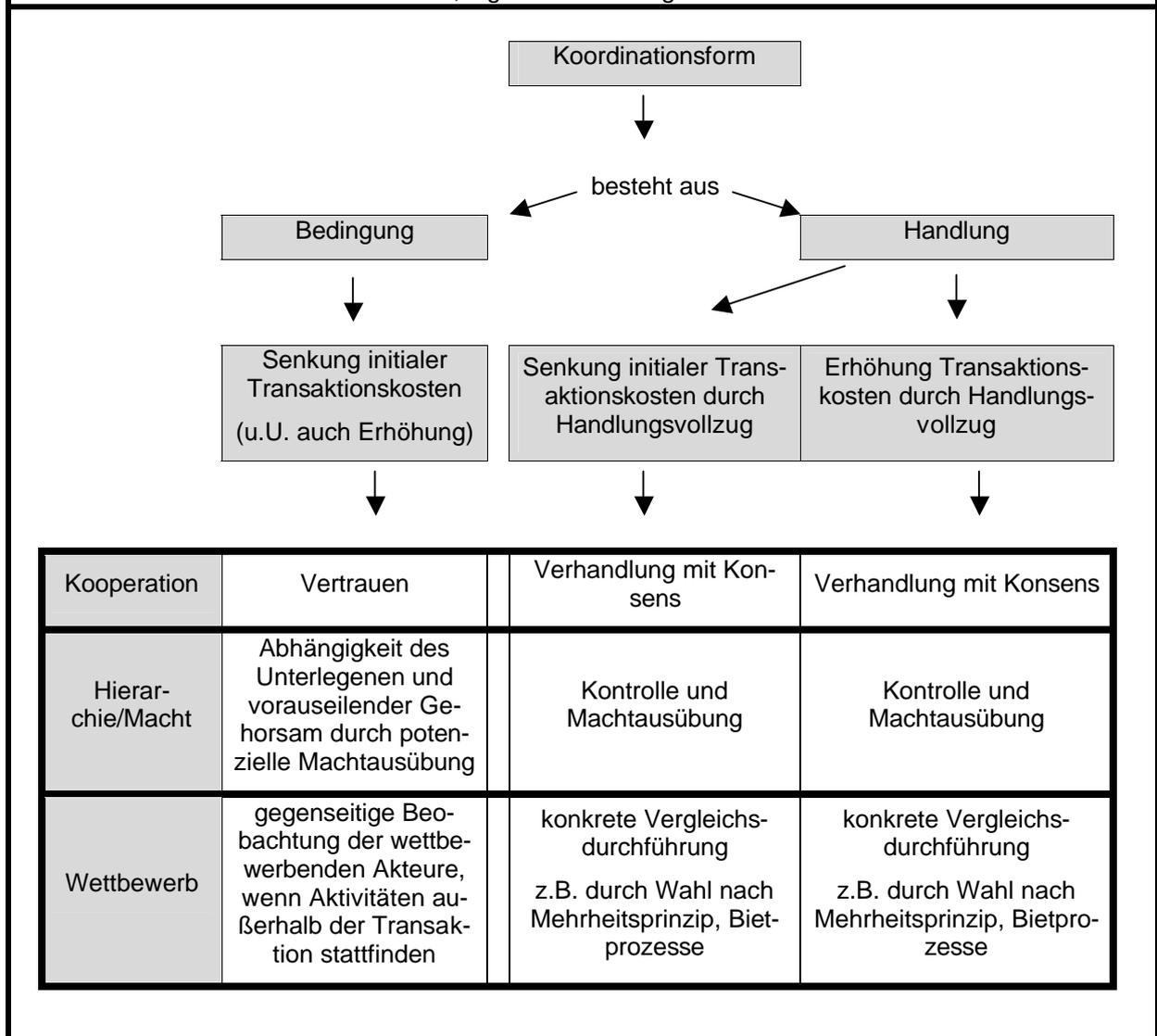
---

<sup>144</sup> Handlungsbedingung und Handlung bilden zusammen als Teile einer Koordinationsform ein Instrument zur Senkung der initialen Transaktionskosten.

ten profitieren. Damit sind alle Beteiligten bei der Win-Win-Situation transaktionstragende Akteure. Nur bestimmte Akteure werden bei den anderen Koordinationsformen zur Transaktion gezählt. Beim Wettbewerb besteht eher eine „Win-Lose“- oder „Winner-takes-all“-Situation, wobei einer die Gesamtnutzenerhöhung erhält, die anderen geschädigt werden oder nur eine anteilige, geringere Nutzenerhöhung erhalten. Bei Hierarchie/Macht können dagegen alle Nutzenszustände zwischen einer Win-Win-Situation wie bei einer Kooperation und einer „Winner-takes-all“-Situation wie beim Wettbewerb auftreten (s.a. Becher et al. 2007; Serries 2005). Bei Hierarchie/Macht und Wettbewerb werden daher im Zusammenhang mit der „Winner-takes-all“-Situation nicht alle Akteure als transaktionstragend gesehen (vgl. Kap. 3.3.). Je nachdem, wie viele (transaktionstragende) Akteure mit ihren Aktivitäten zur Transaktion gezählt werden, sind die Transaktionskosten höher oder niedriger.

Nicht nur über die unterschiedliche Zugehörigkeit zur Transaktion, sondern auch bedingt durch die Handlungsbedingung und Handlung einer Koordinationsform als Instrument können Transaktionskosten gesenkt werden (s. Abb. 6-3). Wenn eine Handlungsbedingung einer Koordinationsform auftritt, sinken die initialen Transaktionskosten, wobei gleichzeitig keine Instrumentnutzungskosten entstehen. Da bei den Handlungsbedingungen keine Handlungen durchgeführt werden, fallen keine Instrumentnutzungskosten an. Wie stark die initialen Transaktionskosten sinken, hängt dabei von der Stärke des Auftretens einer Handlungsbedingung ab. Wenn weitere initiale Transaktionskosten bestehen, können durch die Handlungen einer Koordinationsform die initialen Transaktionskosten weiterhin gesenkt werden. Je mehr initiale Transaktionskosten in Abhängigkeit des Auftretens der Handlungsbedingung gesenkt werden, desto weniger transaktionssenkende Handlungen der Koordinationsformen müssen durchgeführt werden. Beispielsweise senkt bei Kooperation die Handlungsbedingung Vertrauen initiale Transaktionskosten, da über Vertrauen auf frühere Abmachungen oder allgemeine Normen rekurriert wird. Wenn ein hohes Vertrauen vorherrscht, wird vielen Abmachungen und Normen gefolgt, was initiale Transaktionskosten senkt, und Verhandlungen werden kaum benötigt. Zusätzlich steigen die Instrumentnutzungskosten bei Handlungen einer Koordinationsformen an, wenn Handlungen einer Koordinationsform initiale Transaktionskosten senken. Dabei erhöhen sich die Instrumentnutzungskosten proportional zum Auftreten und Intensität der Handlungen. Wenn beispielsweise bei Hierarchie/Macht durch Anreize bei gleichzeitiger Kontrolle der Unterlegenen ein initiales Transaktionsproblem gemildert wird, erhöhen sich die Instrumentnutzungskosten mit dem Vergabeaufwand der (monetären) Anreize und dem Aufwand der Kontrolldurchführung.

**Abb. 6-3:** Unterschiedlicher Einfluss von Kooperation, Hierarchie/Macht und Wettbewerb als Koordinationsformen auf die Transaktionskosten, eigene Darstellung



Manche Handlungsbedingungen und Handlungen bringen auch Nachteile mit sich oder beinhalten Grenzen, die auch mit einer Transaktion und ihren situativen Teilbereichen zusammenhängen, d.h. die Mechanismen bzw. Regeln der Koordinationsformen können in bestimmten Zusammenhängen unpassend sein. Sie wirken nicht nur transaktionskostensenkend, sondern auch –erhöhend, so dass verschiedene Elemente der Koordinationsformen in verschiedenen Situationen kostenminimal sein können (s.a. Noteboom 2002). Sie können also in einer Situation durch eine ausschließliche, dauerhafte und häufige Verwendung hohe Kosten nach sich ziehen, so dass andere Koordinationsformen mit ihren Mechanismen transaktionskostengünstiger sind. Dies kann auch als ein „Versagen“ einer Koordinationsform bzw. ihrer Mechanismen verstanden werden. Dabei kann keine Koordinationsform schädigendes Verhalten vollkommen verhindern, da wegen eines abnehmenden Grenznutzens keine Koordinationsform vollständige Informationen generieren kann. So besteht immer die Möglichkeit, dass Vertrauen missbraucht, eine Kontrolle hintergangen und ein Vergleich manipuliert wird.

Nach diesem ersten Abschnitt wird in einem zweiten Abschnitt der Einfluss der jeweiligen Koordinationsform auf die oben differenziert dargestellten initialen Transaktionsprobleme einer Softwareentwicklung als Transaktion untersucht. Die virtuelle Softwareentwicklung steht dabei im Vordergrund und eine virtuelle Gruppe im Hintergrund. Bei dieser theoretischen Vorüberlegung wird der konkrete Einfluss jeweils einer Koordinationsform auf die Governance- und Motivationsprobleme in Form der Bereitstellungs-, Such-, Mess- und Spezifitätsprobleme vorgestellt. Es wird dabei von einer rein effizienten Koordinationsform ausgegangen, die ohne andere Koordinationsformen wirkt. Für jedes initiale Transaktionsproblem wird nacheinander der Einfluss der Handlungsbedingung und Handlung dieser Koordinationsform analysiert. Dabei können alle drei Koordinationsformen mit ihren Mechanismen und Regeln die initialen Transaktionsprobleme nicht vollständig beseitigen, sondern nur mehr oder minder stark mildern. Die initialen Transaktionskosten können auch erhöht werden. Damit wird zum einen gezeigt, dass Koordinationsformen überhaupt einen Einfluss auf initiale Transaktionsprobleme haben. Zum anderen handelt es sich hierbei um einen logischen Zwischenschritt, der den Einfluss der Koordinationsformen im Open Source-Projekt verständlicher machen soll.

### Wirkungsweise von Kooperation

Eine Kooperation kann entstehen, wenn mögliche gemeinsame Interessen in einer Transaktion signalisiert werden. Dies kann der Fall sein, wenn zuvor überwiegend divergierende, aber auch gemeinsame Interessen bestanden. Nachdem die Kooperationspartner ihre möglichen gemeinsamen Interessen signalisiert haben, wird normalerweise eine Verhandlung angestrebt.

Dabei spielt das gegenseitige Vertrauen als *Handlungsbedingung* der Kooperation eine Rolle, das initiale Transaktionsprobleme verhindert bzw. initiale Transaktionskosten senkt. Das gegenseitige Vertrauen basiert auf dem Signalisieren von gemeinsamen Eigenschaften z.B. einer Gruppe bei eigenschaftsbasiertem Vertrauen, der Vertrautheit im Sinne von vorhandenem prozessbasiertem/kalkulativem Vertrauen und den gesellschaftlich akzeptierten, nicht-personenbezogenen Symbolen wie das Logo einer Organisation, die Absendeadresse einer Email (s.a. institutionenbasiertes Vertrauen) (s. Kap. 3).<sup>145</sup> Es wird dabei postuliert, dass gemeinsame Eigenschaften mit gemeinsamen Interessen einhergehen. Je höher die Ähnlichkeit zum Vertrauensgeber ist, desto höher ist das Vertrauen in einen Akteur und desto stärker werden die initialen Transaktionskosten gesenkt. Bei der Kooperation lösen Vertrauen und Vertrauenswürdigkeit Agency-Probleme somit besonders kostengünstig gegenüber anderen Mechanismen (vgl. Göbel 2002: 123ff.; Ripperger 1998).<sup>146</sup> Mit zunehmendem Vertrauen nimmt Opportunismus ab, verschwindet aber nicht vollständig. Gegenseitiges Vertrauen kann also das Auftreten von initialen Transaktionsproblemen aufheben bzw. mildern, aber nur bis zu einem gewissen Grad (Noteboom 2002: 139ff.). Wird ein Zuviel an Vertrauen gegeben, führt dies zu blindem Vertrauen. Da einseitiges

---

<sup>145</sup> Alle drei Vertrauensarten beziehen sich auf Bekanntschaftsnetzwerke bzw. Bekanntheit in Gruppen, die unterschiedlich groß sein können. Nur das prozessbasierte/kalkulative Vertrauen kann von einem Akteur durch Risikoabschätzungen/Wahrscheinlichkeitsberechnungen bestimmt werden. Bei den beiden anderen Vertrauensarten wird Vertrauen wie das Urvertrauen ohne große Kalkulation geschenkt (vgl. Williamson 1993).

<sup>146</sup> Der Vollständigkeit halber sei hier erwähnt, dass Kooperation in intransparenten Beziehungen wie bei den Freimaurern oder der Mafia dem Vorteil von Teilgruppen oder -projekten dienen kann (Boltanski, Chiapello 2003: 191ff.). Wenn diese Teilgruppen oder -projekte für die Gesamtgruppe oder das Gesamtprojekt schädigende Ziele verfolgen, wird ein Agency-Problem zwischen Gesamt- und Teilgruppe bzw. -projekt nicht gelöst, sondern u.U. erhöht. Da aber die Entscheidungsfindung im untersuchten Open Source-Projekt weitestgehend auf Mailinglisten transparent und nachvollziehbar ist, tritt diese Problematik nicht auf.

Vertrauen auftreten kann, kann bei blindem Vertrauen dieses besonders ausgenutzt werden. Es können Schädigungen für einen Akteur z.B. durch Hold Ups auftreten, weil entgegengesetzte Interessen von einem Akteur ausgeblendet werden (vgl. Ripperger 1998: 63ff.). Außerdem kann eine kooperativ-vertrauensvolle Organisationsformenkultur mit bestimmten unveränderlichen Regeln oder Übereinkünften sowie gegenseitigem Vertrauen bestehen. Dieses gegenseitige Vertrauen kann aber so hoch sein, dass sich nur auf die Vertrauenspartner verlassen und Umweltveränderungen nicht wahrgenommen werden. Von außen angestoßene Konflikte werden vermieden, akkumulieren sich und sind immer schwerer beizulegen. Vertrauen und Verhandlungen können entstehendes Misstrauen mit einer Transaktionskostenerhöhung nicht mehr verhindern und andere Koordinationsformen sind effizienter (vgl. Noteboom 2002: 144f.).

Bei Betrachtung der *Handlung* ist ein umfangreiches Verhandeln dann nicht nötig, wenn ein Signal gemeinsame Interessen bis hin zu einem vollständigen Konsens impliziert und sofortiges Vertrauen möglich macht (vgl. Ciborra, Olson 1988: 95; Noteboom 2002; Möllering 2001). Dadurch besteht kein Problem des Informationsparadoxons, da gegenseitiges Vertrauen die Weitergabe von Informationen über die reziproke oder gruppenorientierte Kooperationsart ermöglicht. Bei diesem Vorvertrauen sind also gemeinsame Interessen bekannt, es sind aber auch noch unbekanntere Interessen vorhanden, die für eine effektive bzw. effiziente Kooperation geklärt werden müssen. Folglich werden gemeinsame Interessen in Form einer Win-Win-Situation angestrebt, aber ein Rest von divergierenden Interessen kann bestehen bleiben, der durch Verhandlungen mit Konsens oder Kompromissen weiterhin beseitigt werden kann. Bei Kompromissen ist eine Nutzenerhöhung für alle Beteiligten vorhanden, es müssen aber Abstriche bei den Maximalforderungen gemacht werden. Die gemeinsamen überwiegen gegenüber entgegengesetzten Interessen, so dass auch bei einem Kompromiss am Ende einer Verhandlung ein Konsens steht. Unabhängig davon, ob ein Konsens über Überzeugen oder ein Kompromiss erreicht wird, erfordern Verhandlungen mit Konsens, die Entscheidungen herbeiführen, hohe Transaktionskosten.

„Tatsächlich werden Verhandlungssystemen in der Literatur schon seit langem hohe Transaktionskosten im Sinne von Entscheidungskosten zugeschrieben, deren Ursache darin gesehen wird, dass eine Verhandlungslösung den Konsens beziehungsweise das Einvernehmen aller Beteiligten verlangt“ (Altemeier 1999: 33).

In einigen Fällen können also Kompromisse nur über langwierige und schwierige Verhandlungen erreicht werden, die ineffizient sind. Dabei kann z.B. die gemeinsame Verfolgung eines Ziels für alle vorteilhaft sein, aber über den jeweilig zu leistenden Beitrag oder die Verteilung des Gewinns herrscht Konflikt. Als Folge können bei der Einigung und dem Ausräumen entgegengesetzter Interessen die Kosten den Nutzen der Kooperation übersteigen. Nachdem ein Kompromiss mit überwiegend gemeinsamen, aber auch entgegengesetzten Interessen ermöglicht wurde, können Konflikte nach Abschluss/Konsensfindung wieder aufbrechen (s.a. „faule“ Kompromisse), was in der Theorie unter Nachverhandlungen bei unvollständigen Verträgen verstanden wird (vgl. Scharpf 1992a, b).

Zusätzlich sind nicht alle gemeinsamen Interessen der Akteure von vornherein klar offengelegt. In diesem Fall kann auf Konsens ausgerichtetes Verhandeln die gemeinsamen Ziele herausarbeiten und Missverständnisse aufdecken. Auf der einen Seite werden dadurch unvollständige Informationen für die Akteure abgebaut und Abmachungen getroffen, auf der anderen Seite entstehen Verhandlungskosten, die mit der Häufigkeit und Dauer der Verhandlungen und Anzahl der Diskutierenden ansteigen. Besteht eine nicht bekannte, aber vorhandene hohe Übereinstimmung gemeinsamer Interessen, dann sind kaum aufwändige Verhandlungen zu erwarten. Sind die Interessen eher entgegengesetzt, ist mit einer länger andauernden Kompromissuche mit hohen Transaktionskosten zu rechnen (vgl. Scharpf 1992a, b; Altemeier 1999: 33). Die Vorbereitungen für eine Verhandlung wie Meinungsgenerierung erfolgen bei den einzelnen Akteuren und spielen

Meinungsgenerierung erfolgen bei den einzelnen Akteuren und spielen daher als Kosten in diesem Fall keine Rolle.

Verhandeln und Vertrauen sind dabei in das jeweilige Kooperationsinstrument eingebettet, d.h. die Kooperationsinstrumente beinhalten Vertrauensarten (s. Kap. 3.2.3.4.), die Einfluss auf das Verhandeln nehmen und Instrumentnutzungskosten senken. Dabei wird auf die Dauer der Verhandlungen, Anzahl der Diskutierenden und Interessensdivergenzen eingewirkt. Die folgenden Kooperationsinstrumente strategische, gruppenbezogene und reziproke Kooperation beinhalten verschieden hohes Vorvertrauen bzw. vorher ausgehandelte, gemeinsame Interessen, die die Entstehung der initialen Transaktionsprobleme verhindern.

Bei der strategischen Kooperation besteht ein gemeinsames Interesse für eine bestimmte Situation oder kurze Dauer, wobei das gemeinsame Interesse genau spezifiziert werden kann. Es besteht ein hohes institutionenbasiertes Vertrauen z.B. bei Spot Markt-Beziehungen im Supermarkt, weswegen kaum Kommunikations-, Austausch- und Verhandlungskosten entstehen, aber initiale Transaktionskosten gesenkt werden.<sup>147</sup> Wenn kein flankierendes institutionenbasiertes Vertrauen besteht, wird vor einer strategischen Kooperation kalkulatives Vertrauen aufgebaut, erfordert aber sehr hohe Kommunikations-, Austausch-, Beobachtungs- und Verhandlungskosten. Initiale Transaktionskosten werden eher wenig reduziert, da nur punktuelle strategische Kooperationen bestehen.

Bei der gruppenbezogenen Kooperation besteht über die gemeinsame Gruppenkultur und ein gemeinsames Grundverständnis Vertrauen gegenüber der Gruppe, wodurch initiale Transaktionskosten gesenkt werden und geringe Kommunikations-, Tauschkosten anfallen.<sup>148</sup> Zusätzlich wird die Suche und Auswahl in einer Gruppe wegen der begrenzten Anzahl an Mitgliedern eingeschränkt. Durch die Nutzung gemeinsamer Ressourcen und die Gabe von Gütern an andere Gruppenmitglieder treten positive externe Effekte auf.

Die reziproke Kooperation ist auf die Etablierung und Aufrechterhaltung von sozialen Beziehungen bei geringer Kenntnis des anderen und seiner Interessen auch in der Zukunft ausgelegt, weswegen hier vor allem Verhandlungen, gepaart mit kalkulativem Vertrauen, geringer Senkung initialer Transaktionskosten und hohen Kommunikations-, Austausch- und Verhandlungskosten auftreten.<sup>149</sup>

Damit entstehen bei der strategischen Kooperation mit institutionenbasiertem Vertrauen bzw. durch den Aufbau von kalkulativem Vertrauen kaum Transaktionskosten. Bei der gruppenbezogenen Kooperation kommen durch Gruppenkultur und institutionenbasiertes sowie eigenschaftsbasiertes Vertrauen eher geringe bis mittelhohe Transaktionskosten vor. Bei der reziproken Kooperation liegen dagegen mit kalkulativem Vertrauen hohe initiale Transaktionskosten vor.

---

<sup>147</sup> Die Entstehung der gesetzlichen und informalen Regelungen des kooperativen strategischen Handelns, die zu einem institutionenbasierten Vertrauen führen, sind dabei für unsere Transaktion nicht von Interesse.

<sup>148</sup> Die transaktionskostensenkende Wirkung der gruppenbezogenen Kooperation basiert vor allem darauf, dass die Gruppe bereits existiert. Die hohen Transaktionskosten der Gruppenbildung werden nicht berücksichtigt, d.h. gehören nicht zur Transaktion.

<sup>149</sup> Bei der reziproken Kooperationsart existiert die Kooperation u.U. noch nicht oder diese Kooperationsart steht in einer langen, unregelmäßigen Reihe von Interaktionen. Hohe Transaktionskosten bestehen, da oftmals keine Ausgangsbasis vorhanden ist und die gemeinsamen Interessen zuerst erkundet oder fortgeschrieben werden müssen. Da in zukünftigen Treffen eine andere Situation vorherrscht, müssen auch hier die neuen Modalitäten wenigstens z.T. festgelegt werden. Dabei wird alles in der Transaktion ausgehandelt.

Im Folgenden wird der Einfluss auf die initialen Transaktionskosten der *Bereitstellungs- und Suchprobleme* sowie *Mess- und Spezifitätsprobleme* durch Handlungsbedingung und Handlung der Kooperation untersucht.

- Das *Bereitstellungsproblem* wird durch Vertrauen stark gemildert, wenn potenzielle Mitarbeiter z.B. durch Bekanntschaftsnetzwerke bekannt sind und daher einfach in die Softwareentwicklung integriert werden können. Dabei spielt auch ein gemeinsames Verständnis eine Rolle, das die Beteiligten mitbringen und das nicht digitalisiert werden kann. Das Informationsparadoxon oder ein Anonymitätsproblem bzgl. falscher Identität tritt durch das Vorhandensein von mindestens minimalem Vertrauen bei der Übermittlung digitaler Daten nicht auf. Eine Person möchte beispielsweise freiwillig und aus eigenem Antrieb mitarbeiten (intrinsische Motivation) und bringt das geeignete Humankapital mit. Zusätzlich signalisiert diese Person starke Konformität mit den Zielen einer Transaktion. In diesem Fall werden initiale Transaktionskosten stark gesenkt und wenige Instrumentnutzungskosten entstehen. Das Bereitstellungsproblem kann aber auch durch Verhandlungen mit Konsens gelöst werden, wenn potenzielle Mitarbeiter angesprochen und dabei Abmachungen mit Akteuren getroffen werden. Dadurch sind jedoch hohe Verhandlungskosten bei einer hohen Anzahl von Akteuren und einer hohen Verhandlungsdauer zu erwarten. Dies gilt auch für die Digitalisierbarkeit bzw. Bereitstellung von Daten.

- Das *Suchproblem* beinhaltet einen Information Overflow durch die IuK-Technik, der durch Vertrauen in die Information gesenkt werden kann. Da aber auch falsche und irrelevante Informationen im Internet zu finden sind, kann Vertrauen fehlgeleitet und ausgenutzt werden. Weiteres Verhandeln mit angestrebtem Konsens, wobei der Kommunikationspartner Zielkonformität signalisiert, erhöht die Instrumentnutzungskosten, verringert aber zugleich den Information Overflow. Da durch Anonymität kein Zugriff auf den tatsächlichen Akteur besteht, kann eine Verhandlung letztendlich das Adverse Selection-Problem nicht ganz verhindern, jedoch verringern.

- Ein *Messproblem* mit Shirking und Free Rider-Verhalten/Fringe Benefits kann durch gegenseitiges Vertrauen entschärft werden, wenn alle Akteure ihre Interessenkonformität durch die Zugehörigkeit z.B. zu einer spezifischen Gruppe signalisieren. Weiterhin kann ein Akteur von sich aus Informationen weitergeben, sog. Reporting, und somit Vertrauen generieren (vgl. Göbel 2002: 101ff.). Durch die gemeinsame Festlegung gemeinsamer Interessen kann außerdem davon ausgegangen werden, dass Shirking und Free Rider-Verhalten/Fringe Benefits kaum auftreten, wenn die Beteiligten ehrlich zueinander sind. Es existiert aber immer noch eine minimale Chance, dass beide Probleme vorhanden sind und eine Gefahr des Opportunismus besteht.

- Ähnlich wie beim Messproblem kann Vertrauen *Spezifitätsprobleme* nur bedingt abschwächen. Gegenseitiges Vertrauen kann Kosten aus möglichen Hold Ups verringern, die z.B. durch IuK-technikbezogene, geringere Zeitspezifität oder investiertes Humankapital entstehen.<sup>150</sup> Ein Akteur nutzt dabei seine überlegene Situation nicht aus. Aber Verhandlungen mit Konsens erhöhen die Kosten bei gleichzeitiger Milderung der Hold Ups. Dies gilt besonders für Nachverhandlungen eines unvollständigen Vertrags, die auch durch Vertrauen kaum abgeschwächt werden. Es treten kaum Verhandlungskosten auf, wenn nur Teile neu ausgehandelt werden. Die Notwendigkeit zu Nachverhandlungen kann aber umgekehrt das Vertrauen untergraben und zu erhöhten Transaktionskosten führen, da eine angenommene Einigkeit unvorhergesehen nicht mehr existiert. Damit besteht

---

<sup>150</sup> Vertrauen unterstützt zusätzlich die erhöhte Kontaktwahrscheinlichkeit zwischen Gleichgesinnten durch die geringere Standortspezifität. Räumliche Nähe als Standortspezifität unterstützt Verhandlungen, da eine direkte und synchrone Kommunikation möglich ist (s.a. Kap. 6).

auch hier die Gefahr, dass der andere Akteur seine Interessen nicht vollständig offengelegt und sich von vornherein unerkannt opportunistisch verhalten hat.

#### Wirkungsweise von Hierarchie/Macht

Macht tritt in dieser Theorieinterpretation in zwei Formen auf: Erstens kann Macht schädigend sein, wobei sich die Kosten für eine untersuchte Transaktion erhöhen. Zweitens kann Macht, positiv betrachtet, auftreten, einer untersuchten Transaktion nutzen und die Kosten für diese erniedrigen. Die Art und Weise des Machteinsatzes unterscheidet sich nicht, sondern die Ausrichtung des Machteinsatzes gegen das oder mit dem Ziel der Transaktion. An dieser Stelle wird die positive, transaktionskonforme Macht untersucht, die u.U. die transaktionsschädigende Macht verhindern kann.

Die Stärke des Einflusses des Über-/Unterlegenheitsverhältnisses als *Handlungsbedingung* der Hierarchie/Macht auf die initialen Transaktionsprobleme wird von der Abhängigkeit des Unterlegenen bestimmt. Die Höhe der Abhängigkeit eines Unterlegenen bewirkt, dass ein Unterlegener sich konform den Wünschen des Mächtigen verhält, so dass die initialen Transaktionsprobleme nicht auftreten. Diese Abhängigkeit kann z.B. aus Humankapital- oder anderen Investitionen (s.a. Lock In-Effekt) oder aus der Abhängigkeit des Akteurs von den Ressourcen des Überlegenen bzw. Prinzipals entstehen. Üblicherweise wird in der Theorie unter dem Prinzipal z.B. ein Vorgesetzter in einem Unternehmen<sup>151</sup> oder Aktienbesitzer verstanden (vgl. Göbel 2002). Bei der Softwareentwicklung kann der Prinzipal durch einen oder mehrere Beteiligte u.a. mit formalen Posten vertreten sein, z.B. als Projektleiter.

Außerdem bewirkt eine potenzielle Machtdurchführung in Form einer drohenden Sanktionierung, von Versprechen (materieller) Anreize oder von manipulativer Macht eine Verhaltenssteuerung, was auch auf eine Transaktionskostensenkung bei den Transaktionsproblemen hinausläuft. Je stärker also die Abhängigkeit bei gleichzeitiger Angst vor Strafe bzw. Erwartung einer möglichen Belohnung oder beim Zugang zu Informationen (manipulative Macht), desto stärker richtet der Agent sein Verhalten an dem Willen des Überlegenen aus. Dies senkt initiale Transaktionskosten, da z.B. Informationen im vorausseilenden Gehorsam weitergegeben werden (s.a. Reporting)<sup>152</sup>. Umgekehrt wird eine Machtausübung nicht oder kaum benötigt, was Instrumentnutzungskosten senkt.

Die *Handlung* umfasst Machtausübung und Kontrolle und hängt mit dem Nutzen für den Prinzipal zusammen. Da die Nutzenmaximierung des Prinzipals/Überlegenen bei der Hierarchie/Macht immer maximiert wird, sollte diese im Sinne der Transaktion sein, also der Softwareentwicklung. Damit unterstützt Hierarchie/Macht die Transaktionsdurchführung. Wenn der Prinzipal seinen Nutzen erhöht und dabei die Transaktionsdurchführung gefährdet oder verteuert, ist dies aus Sicht der Transaktion unerwünscht und führt zu einer Transaktionskostenerhöhung. Die gleichzeitige Nutzenerhöhung des Prinzipals und des Agenten bezieht sich auf eine Belohnung (bzw. Anreize) für den/die Agenten, ist im Sinne der Transaktion und senkt dabei gleichzeitig Transaktionskosten. Die fehlende Nutzenmaximierung bzw. die Schädigung des Unterlegenen/Agenten bezieht sich auf eine Bestrafung des/der Agenten, der die Transaktion schädigt.

---

<sup>151</sup> Wie schon oben beschrieben, ist der Arbeitnehmer in einem Unternehmen gegenüber dem Arbeitgeber bzw. Vorgesetzten als Arbeitgebervertreter unterlegen, auch wenn er zu einem „freiwilligen“ Arbeitsverhältnis einwilligt, weil er von den Lohnzahlungen abhängig ist. Das heißt aber nicht, dass der Arbeitnehmer wehrlos wäre. Durch die Informationsasymmetrie zwischen Arbeitgeber und Arbeitnehmer kann letzterer Gegenmacht z.B. durch einen Hold Up aufbauen.

<sup>152</sup> Reporting an sich sagt nicht aus, ob die Information wegen einer Angst vor Strafe oder aus Kooperation weitergegeben wird.

Der Prinzipal kann nicht nur eine Person, sondern auch mehrere Personen bzw. eine Personengruppe sein. Da aber der Überlegene bzw. die Überlegenen meist in der Minderzahl sind und nicht alle Beteiligten, darunter auch die Unterlegenen/Agenten, beteiligt sind, fallen kaum Kosten für Entscheidungen an. Bei einer Person als Prinzipal entstehen bei Entscheidungen sogar kaum oder gar keine Kosten. Da hauptsächlich virtuell kommuniziert wird, tritt vor allem kommunikative, textbasierte Macht auch als kollektive Sanktion auf.

Die Handlung der Hierarchie/Macht beinhaltet oft eine Willensbekundung des Überlegenen, die aus einer konkreten Drohung, möglichen Belohnung oder Anweisung über eine mögliche nachfolgende Machtdurchführung beim Prinzipal besteht. Zudem können noch Entscheidungskosten hinzugerechnet werden (s.o.). Im Falle der manipulativen Macht entstehen eher geringe Kommunikationskosten, weil dieses Machtinstrument vor allem auf Kommunikation basiert. Wird nur auf manipulative Macht gesetzt, kann der Überlegene sein Interesse verschleiern und diese Art von Kommunikationskosten treten nicht auf. Kosten kommen hinzu, wenn aktive Maßnahmen ergriffen werden und die Drohung durch Bestrafen umgesetzt wird. Wenig bis keine Kosten entstehen für den Überlegenen, wenn dieser dem Unterlegenen den Zugang zu Ressourcen als Sanktion verwehrt, von denen der Unterlegene abhängig ist, um seine Ziele wie Softwareentwicklung zu verfolgen.<sup>153</sup> Dabei erhöht aber der Überlegene die Kosten des Unterlegenen, der seine Ziele zu verfolgen sucht. Werden Anreize ein- und physische Gewalt umgesetzt, treten zusätzlich die Anreizkosten/Durchführungskosten auf. Bei der Verwendung zusätzlicher manipulativer Macht z.B. Einsatz von Expertenwissen oder Informationsmacht entstehen weitere Kommunikationskosten. Die Kosten steigen mit der Anzahl der Personen, Frequenz und Dauer an.

Die Kontrolle, ob der Unterlegene nach dem Willen des Überlegenen handelt, ist direkt mit dem Machteinsatz verbunden. Bei Hierarchie/Macht senkt die Kontrolle die Transaktionskosten der initialen Transaktionsprobleme, indem diese für den Mächtigen bzw. Prinzipal Informationen über den Unterlegenen/Agenten generiert und so das Auftreten unvollständiger asymmetrischer Informationen und eines Informationsparadoxons vermeidet. Andererseits erzeugt die Kontrolle als informationsgenerierendes Element zusätzliche Instrumentnutzungskosten, da eine zu überwindende asymmetrische Informationsverteilung zwischen Kontrolleur und Kontrollierten besteht. Die Kontrolle des Untergebenen generiert Kosten, die mit ihrer Kontrollfrequenz, Dauer der Kontrolle und der Anzahl der zu Kontrollierenden ansteigt. Beispielsweise steigen die Kontrollkosten mit detaillierterer Kontrolle immer stärker an. Bei einer Verhaltenskontrolle (s.a. Monitoring) wird das Verhalten ständig überwacht, was bei anhaltender Dauer akkumulierende hohe Kontrollkosten generieren kann. Je stärker die Kontrolle dabei ist, desto eher lähmt dies jede Initiative in Form von intrinsischer Motivation der Kontrollierten für die Transaktion. Weitere Anreize mit Kosten sind nötig, da ohne ständige Kontrolle mit potenziellem Machteinsatz keine (Zusammen-)Arbeit geleistet wird (vgl. Jung 2005: 450; Sader 1991: 272f.; Rosenstiel et al. 1995: 274ff.). Die Endkontrolle eines fertigen Produkts ist punktuell umfangreich und führt dann kurzzeitig zu hohen Kosten. Die luK-Technik ermöglicht aber eine effektivere Überwachung der Aktivitäten in Form von Änderungskontrollen, da z.B. in einem Versionsmanagementsystem oder einer Datenbank über den Warenbestand das reale Verhalten Spuren in Form von Änderungen im luK-technischen System hinterlässt. Diese Änderungen können dann nach spezifischen (statistischen) Mustern untersucht werden (vgl. Ciborra, Olson 1988; Schmiede 2006: 464). Nachteilig ist dabei, dass ein abnehmender Grenznutzen und

<sup>153</sup> Beispielsweise bei einer Vertragsbeziehung mit anschließenden Geldzahlungen zwischen Privatperson und Handwerker kann Druck zur Korrektur fehlerhafter Arbeit durch die Zurückhaltung von Geld aufgebaut werden.

eine zunehmende Anzahl zu beobachtender Veränderungen eine solche Kontrolle zunehmend kostenintensiv werden lassen. Außerdem besteht eine weitere Unsicherheit, weil nicht alle Aktivitäten vollständig digitalisierbar sind und nicht jedes Verhalten wegen der ansteigenden Kontrollkosten beobachtet werden kann.

Wird bei der Kontrolle festgestellt, dass der Unterlegene sich nicht wie gewünscht verhalten hat, wird wieder Macht eingesetzt. Kontrolle und Machteinsatz können so lange erfolgen, bis der oder die Unterlegenen sich wie gewünscht verhalten, was zu ansteigenden Kosten des Machteinsatzes und der Kontrolle führt.

Die Kosten aus der Kontrolle und der Machtausübung fallen normalerweise beim Mächtigen an und zählen zur Transaktion, solange die Instrumente der Hierarchie/Macht in der Transaktion durchgeführt werden. Es fallen aber u.U. auch Transaktionskosten bei den Unterlegenen an, die sich abhängig von dem Willen des Überlegenen verhalten, aber nicht unbedingt zur Transaktion gezahlt werden können. Geringe Instrumentnutzungskosten in der Transaktion entstehen, wenn nur einer oder wenige Beteiligte Verhaltenweisen sanktionieren und die Kosten der Kontrolle und der Machtausübung nur bei diesen anfallen. Dies ist der Fall, wenn nur ein Mächtiger/Prinzipal als Vertreter einer Gruppe (z.B. des KDE-Projekts) bzw. der Transaktion, im Namen der Mitglieder, die Interessen der Mitglieder gegenüber einem oder mehreren Mitarbeitern durchsetzt. Dabei wird der Nutzen aller Mitarbeiter bei dieser kollektiven Sanktion erhöht bzw. nicht abgesenkt und nur der Nutzen eines schädigenden Mitarbeiters gesenkt (vgl. Jones et al. 1997). Im extremsten Fall wird der Mitarbeiter ausgeschlossen und verliert seine Investitionen in die Beziehung zu den anderen Mitarbeitern bzw. in die Software.

Außerdem spielt im Falle der Softwareentwicklung wegen der hohen Virtualität die physische Gewalt keine Rolle, d.h. die Kosten einer Durchführung wären immens hoch. Kaum Transaktionskosten verursacht der Entzug von Ressourcen, von denen der Unterlegene abhängig ist, wenn dies kommentarlos bleibt. Kommunikationskosten treten aber auf, wenn dieser Entzug erklärt wird.

Bei einer ständigen und starken Kontrolle „bindet sich“ der Kontrolleur an sein Kontrollobjekt und kann keine anderen Tätigkeiten nebenher ausführen. Diese anderen Kontrollen durchführen zu können, erfordert daher weitere Transaktionskosten. Erfolgt eine ständige Kontrolle mit ständigem Machteinsatz, entstehen bei den Unterlegenen Aggressionen, die sich gegen andere Beteiligte oder den Überlegenen richten und zum Gebrauch von Gegenmacht führen können. Ein Austritt aus einer Gruppe (wie dem KDE-Projekt) wäre auch denkbar. Dabei gehen bei der Softwareentwicklung Investitionen z.B. in Humankapital o.ä. verloren, die nur durch hohe Kosten wieder aufgebaut werden können (vgl. Sader 1991: 272f.; Rosenstiel et al. 1995: 274ff.).

Handlungsbedingung und Handlung der Hierarchie/Macht senken die initialen Transaktionskosten aus den folgenden *Bereitstellungs- und Suchproblemen sowie Mess- und Spezifitätsproblemen*.

- Kosten aus einem *Bereitstellungsproblem* werden durch die Angst vor Strafe oder mögliche Belohnungen gesenkt, da die abhängigen, gesuchten Personen sich im vorausgehenden Gehorsam melden. Auch das Informationsparadoxon wird durch Angst vor Strafe oder durch mögliche Belohnungen umgangen, da die Unterlegenen freiwillig Informationen weitergeben. Um einen solchen Gehorsam zu erreichen, muss vorher eine Willensbekundung des Mächtigen bekannt sein, dass Informationen (wie Quellcode) oder Mitarbeiter benötigt werden. Durch die Willensbekundung entstehen nur minimale Kommunikationskosten und geringe Kontrollkosten des Verhaltens, da die Unterlegenen nicht durch einen konkreten

Machteinsatz zu einem Verhalten angeleitet werden, sondern sich vorausschauend auf das Gewollte ausrichten. Die Kosten können durch den Einsatz von Kontrolle und Macht weiter gesteigert werden, wenn keine Informationen oder Mitarbeiter der Willensbekundung gemäß bereitstehen.

- Bei *Suchproblemen* treten Information Overflows und Adverse Selection auf, die über Angst vor Strafe, mögliche Belohnungen oder manipulative Macht, wie Drohungen oder manipulierte Informationen beseitigt werden. Unterlegene geben daher wie beim Bereitstellungsproblem nur korrekte und passende Informationen weiter. Wenn diese potenzielle Macht nicht wirkt, kann ein konkreter Machteinsatz, wie Bestrafen, Belohnen und Manipulation<sup>154</sup>, mit Kontrolle zudem die Feststellung der korrekten und adäquaten Informationen ermöglichen. Dabei steigen mit der Anzahl der Personen, der Dauer und Häufigkeit die Instrumentnutzungskosten.

- Ein *Messproblem* bezieht sich bei der virtuellen Softwareentwicklung auf Shirking und Free Rider-Verhalten/Fringe Benefits. Diese Probleme treten wie oben erwähnt wegen der Angst vor Strafe, möglichen Belohnungen oder manipulativer Macht kaum auf. Da aber ein potenzieller Machteinsatz diese Probleme nicht vollständig verhindern kann, werden Bestrafen, Belohnen und manipulative Macht mit Kontrolle zur weiteren Reduktion initialer Transaktionskosten herangezogen. Das Messproblem kann durch Kontrollen überwunden werden. Eine Verhaltenskontrolle kann aber wegen der standortungebundenen Telearbeit durch die IuK-Technik nicht unbedingt durchgeführt werden. Eine Endkontrolle des fertigen Softwareprodukts und eine Kontrolle aller auftretenden Veränderungen kann das Messproblem umgehen. Eine Endkontrolle birgt jedoch immer das Risiko, dass das fertige Softwareprodukt z.B. durch äußere, nicht erkennbare Umwelteinflüsse nicht den formalen bzw. informalen Ansprüchen des Auftraggebers genügt und weitere Verbesserungskosten eingesetzt werden müssen (s. Ebers, Gotsch 1999: 214f.; Göbel 2002: 115f.).

- Ein *Spezifitätsproblem* hängt mit Hold Ups zusammen, die die Transaktion schädigen. Diese gehen daher bei Hierarchie/Macht in der Softwareentwicklung von einem Agenten aus. Spezifitätsprobleme können durch potenzielle Machteinsätze abgeschreckt und zusätzlich durch reale Machteinsätze mit Kontrollen, aber ansteigenden Kosten gesenkt werden. Hold Ups, die vom Prinzipal ausgehen, produzieren Kosten, die aber den Nutzen für die Transaktion erhöhen. Bei Nachverhandlungen von Verträgen kann der Überlegene seine Interessen eher durchsetzen.

### Wirkungsweise von Wettbewerb

Die *Handlungsbedingung* des Wettbewerbs beinhaltet, dass sich die Wettbewerbbenden mit den Wettbewerbsbedingungen/-merkmalen oder mit anderen Beteiligten vergleichen können. Die Wettbewerber versuchen, ihre Merkmale an diese Wettbewerbsmerkmale so optimal wie möglich anzupassen, d.h. besser als der durchschnittliche bzw. beobachtete Wettbewerber zu sein (s.a. Kap. 3). Dabei müssen am Anfang eines Wettbewerbs ein oder mehrere transaktionstragende Akteure den Wettbewerb z.B. durch eine Ausschreibung bekannt machen und/oder die Wettbewerbsbedingungen und -merkmale festlegen, was minimale Transaktionskosten nach sich zieht. Dabei können die Wettbewerber die anderen Wettbewerber beobachten und ihre Aktionen auf deren Aktionen abstimmen, wobei zusätzlich Anpassungskosten bei diesen anfallen. Wenn der Wettbewerb formal bekannt gemacht wurde, treten eher nur relativ geeignete Wettbewerber freiwillig in den Wettbe-

---

<sup>154</sup> Die Verringerung von initialen Transaktionskosten über manipulative Macht erfolgt über die Weitergabe von spezifischer Information, sich Anbieten als Identifikationsfigur und das Auftreten als Experte, der den Wahrheitsgehalt einer Information sofort feststellen kann.

werb ein (s.a. Self Selection). Durch diese Vorauswahl sinken die initialen Transaktionskosten, wobei die Instrumentnutzungskosten des Vergleichs eher geringer ausfallen.

Eine zusätzliche Senkung initialer Transaktionskosten erfolgt, da die Wettbewerber Informationen über einander generieren und ihre Wettbewerbsmerkmale anpassen und optimieren. Sie entfernen sich zwar vom Zustand der unvollständigen Information, aber dafür entstehen Vergleichs-, Beobachtungs-, Such- und Bewertungskosten von Mitbewerbern sowie ständige Anpassungskosten an neue Bedingungen.<sup>155</sup> Ohne wechselseitige Beobachtung vergleichen sich die Akteure nur mit den gegebenen Anforderungen, was zu einer geringeren Senkung der initialen Transaktionskosten und geringen Vergleichs- bzw. Anpassungskosten bei den Wettbewerbern führt.

Die Wettbewerber können von der Transaktion aus gesehen zu dieser gerechnet werden, aber auch außerhalb der Transaktion stehen. Dabei gehen die Anpassungskosten der Wettbewerber in die Berechnung der Transaktionskosten ein oder nicht ein. Die transaktionstragenden Akteure stehen z.B. in einer Softwareentwicklung in einem Wettbewerb um Lösungsmöglichkeiten, wobei die Kosten der Beteiligten zur Transaktion gezahlt werden und hohe Instrumentnutzungskosten anfallen. Umgekehrt sinken die initialen Transaktionskosten durch alle Wettbewerber, die sich übereinander und das angestrebte Ziel informieren, wobei aber nur einer oder eine geringe Anzahl von Wettbewerbern zur Transaktion gerechnet wird. Die Anpassungskosten der restlichen Wettbewerber fallen dann nicht in der untersuchten Transaktion an und werden deswegen bei der Betrachtung nicht zur Transaktion gezählt.

Es gibt auch wettbewerbles Verhalten, das der Transaktion nicht nützt bzw. diese schädigt. Folgende in Gruppen eher seltene Problematik kann aber u.U. einen großen Einfluss haben: Ein Austritt oder Abgang aus einem Wettbewerb bei der Softwareentwicklung kann die Transaktion schädigen, da Wissen und Arbeitskraft abhanden kommt. Dies wird bei einem direkten, ruinösen Wettbewerb angestrebt, bei dem eine Person den eigenen Nutzen kurzfristig verringert, um einen größeren Nutzenverlust bei anderen und später dadurch eine starke eigene Nutzenerhöhung zu erreichen, d.h. es wird versucht, Mitbewerber aus dem Wettbewerb zu drängen (s. Kap. 3.2.3.6.). Bei einem Kampf bzw. verbalen Konflikt tritt dies auf, wobei die direkte (Ruf-)Schädigung des Gegners das Ziel ist. Bei einem Ausscheiden eines Wettbewerbers verliert dieser Anpassungskosten, die als Investitionen in eine bestehende oder zukünftige (Vertrags-)Beziehung gesehen werden können und damit in Zusammenhang mit einem Spezifitätsproblem stehen. Die transaktionstragenden Akteure dagegen verlieren dadurch einen u.U. passenden Wettbewerber als Vertragspartner, obwohl die Vergleichskosten als Mechanismus sinken können (s.u.).

Transaktionskosten entstehen durch die Durchführung eines Vergleichs als *Handlung* eines Wettbewerbs durch einen oder mehrere transaktionstragende Akteure und fallen daher bei der Transaktion an. Eine Senkung der initialen Transaktionskosten bei hohen Instrumentnutzungskosten erfolgt durch den Vergleich der Merkmale von Wettbewerbern, wobei aktiv von dem oder den transaktionstragenden Akteuren Informationen über diese eingeholt und verglichen werden (s.a. Screening). Demgegenüber können die Instrumentnutzungskosten eines Vergleichs bei einer Ausschreibung geringer sein, obwohl eine Ausschreibung Kosten des Bekanntmachens und der Festlegung der Wettbewerbsmerkmale beinhaltet (s.o.). Die Wettbewerber präsentieren in diesem Fall gegenüber dem oder den Wettbewerbsausrichter(n) die eigenen Merkmale (s.a. Signalling).

---

<sup>155</sup> Als mögliche Folge sinken besonders die initialen Transaktionskosten, die auf Informationsdefiziten beruhen.

Der Vergleich verringert generell Informationsasymmetrien, weswegen das Informationsparadoxon mit der anwachsenden Anzahl der alternativen Informationsmöglichkeiten vermindert wird. Es werden aber nur die Merkmale verglichen, die direkt auf den Wettbewerb bezogen sind. Dies reduziert die Kosten eines Vergleichs, lässt aber viel Freiraum für weitere auch transaktionsschädliche Interessen. Die Merkmale eines Vergleichs müssen daher sorgfältig ausgewählt werden, da sonst Schädigungen möglich sind. Die Instrumentnutzungskosten eines Vergleichs steigen an, wenn nicht einer sondern mehrere transaktionstragende Akteure den Vergleich durchführen. Dabei kann eine Wahl nach dem Mehrheitsprinzip eine Entscheidung bei vielen Beteiligten herbeiführen.

Unabhängig davon, wie der Vergleich durchgeführt wird, steigen die Kosten des Vergleichs auf der einen Seite, wie beim Mehrheitsprinzip bekannt, mit der Anzahl der Alternativen stark an. Dies lässt mit der Anzahl der Vergleichsmerkmale die Kosten weiter steigen. Auf der anderen Seite erhöhen sich die Kosten einer demokratischen Mehrheitswahl mit der Anzahl der Beteiligten nur leicht (vgl. Sartori 1984, 1992). Auch steigen die Vergleichskosten mit der Komplexität des Vergleichsverfahrens, können aber durch die Automatisierung des Vergleichsverfahrens durch die IuK-Technik wieder gesenkt werden. Wenn ein Vertreter allein durch einen Bietprozess z.B. anhand des gebotenen Preises wie in einer Auktion entscheidet, beinhaltet das Vergleichsverfahren eher wenige Instrumentnutzungskosten. Bei einer demokratischen Mehrheitswahl dagegen beobachten und beeinflussen sich mehrere transaktionstragende Akteure, wobei deren Transaktionskosten mit den Kosten der Mehrheitswahldurchführung in hohen Instrumentnutzungskosten resultieren.

Nach der Durchführung eines Vergleichs als Mechanismus zieht nur ein Beteiligter den maximalen Nutzen aus dem Wettbewerb. Die Mitwettbewerber haben hohe Kosten, aber keinen Nutzen, somit verpufft die Motivation der Mitwettbewerber. Die demotivierten Beteiligten verlassen die Gruppe/Softwarezusammenarbeit bei fehlenden weiteren sozialen Bindungen, was ansteigende Werbungskosten neuer Mitarbeiter für die Gruppe nach sich zieht.

Die Handlungsbedingung und Handlung des Wettbewerbs beeinflussen die folgenden *Bereitstellungs-, Such-, Mess- und Spezifitätsprobleme*.

- Bei einem *Bereitstellungsproblem* entstehen generelle Kosten der Etablierung eines Wettbewerbs durch die Bekanntgabe einer Wettbewerbshandlung in Form einer Ausschreibung, eines Wettkampfs o.ä. Die Kosten des Informationsparadoxons werden vor allem durch die gegenseitige Beobachtung der Wettbewerbenden gesenkt, die dadurch Informationen über sich gewinnen und ihre Wettbewerbsmerkmale verbessern. Um am Wettbewerb bei virtueller Zusammenarbeit teilnehmen zu können, digitalisieren die Wettbewerber ihre Merkmale freiwillig. Die Durchführung eines Vergleichs wiederum senkt die Kosten aus dem Bereitstellungsproblem bei gleichzeitig stark ansteigendem Vergleichsaufwand.

- Adverse Selection als *Suchproblem* entsteht nicht, wenn die richtigen Wettbewerbsmerkmale ausgewählt wurden, wobei ein Großteil schädigender Interessen ausgeschlossen werden kann. Das initiale Transaktionsproblem wird gemildert, wenn viele Wettbewerbsmerkmale eingefordert werden, so dass ein detaillierteres Bild der Transaktionspart-

ner entsteht.<sup>156</sup> Zugleich entsteht ein Information Overflow, wobei sich die Instrumentnutzungskosten durch den Vergleich vieler Merkmale erhöhen.

Die Wettbewerber verbessern und passen bei einer Ausschreibung ihre Merkmale an die Wettbewerbsmerkmale an, so dass der am besten Geeignete ausgewählt wird. Durch die gute Eignung des Wettbewerbers sind ex post geringe Transaktionskosten zu erwarten. Information Overflows können bei den Wettbewerbern entstehen, die sich gegenseitig beobachten. Ob dies die Transaktionskosten erhöht oder nicht, hängt davon ab, ob die Wettbewerber zur Transaktion gehören.

- Shirking als ein *Messproblem* wird am Besten durch den direkten Vergleich von Arbeitsergebnissen in Form von Arbeits- bzw. Softwarepaketen als Lösungen für eine gleiche oder ähnliche Aufgabe abgeschwächt, da im KDE-Projekt die Erstellungszeit keine Rolle spielt. In anderen Fällen kann ggf. noch die Arbeitszeit mit verglichen werden. Bei einem Vergleich von Absichtserklärungen wie Angeboten kann zwar durch die Auswahl eines Mitarbeiters mit transaktionskonformen Interessen ein Shirking-Problem vermindert, aber nicht vermieden werden. Die einzelnen Wettbewerber optimieren ihre Softwarepakete auf das Geforderte, so dass jeder sein Bestes gibt und zwischen mehreren optimalen Arbeitsergebnissen bzw. Angeboten verglichen werden kann. Die Kosten der Wettbewerber beinhalten aber zusätzlich Opportunitätskosten, da sie während der Erstellung eines auszuwählenden Arbeitsergebnisses oder Angebots auch die Möglichkeit hätten, an anderen Aufgaben zu arbeiten. Die Transaktionskosten steigen, wenn die Wettbewerber innerhalb der Transaktion in einem Wettbewerb stehen. Das Free Rider-Problem/Fringe Benefits tritt bei einem Vergleich von Arbeitsergebnissen kaum auf, kann aber bei Angeboten u.a. eine Rolle spielen. Eine geringere Abschwächung dieses Messproblems erfolgt, wenn nur ein grober Vergleich mit Erfahrungen des Vergleichenden oder ähnlichen, erledigten Arbeitsaufgaben anderer Personen möglich ist. Wird eine Mehrheitswahl z.B. über Entscheidungen zu Arbeitsangeboten oder Quellcode durchgeführt, sind die Instrumentnutzungskosten besonders hoch.

- Die beim *Spezifitätsproblem* möglicherweise auftretenden Hold Ups werden am besten abgeschwächt, wenn wie beim Messproblem verschiedene Arbeitspakete für eine Arbeitsaufgabe verglichen werden.<sup>157</sup> Dabei sinkt die Möglichkeit eines Hold Ups zwischen den Softwareentwicklern und dem Vergleichenden mit der Anzahl der angebotenen Arbeits- bzw. Softwarepakete als Lösung für eine Aufgabe, kann aber bei Angeboten weiterhin bestehen. Auch hier fallen Kosten durch Opportunitätskosten bei transaktionstragenden Wettbewerbern an. Nachverhandlungen treten durch die Auswahl des am besten passenden Arbeitspakets seltener auf, können aber nicht verhindert werden. Auch wenn das am besten passende Arbeitspaket gewählt wird, können Nachbesserungen mit Nachverhandlungen nötig sein oder werden. Daraus kann u.U. ein Hold Up des Softwareentwicklers als Wettbewerber gegenüber dem Vergleichenden abgeleitet werden.

### **6.1.2. Analyse der Koordinationsformen im Open Source-Projekt KDE**

Im Folgenden wird in einer zweiten Stufe eine Anpassung der umrissenen Transaktionskostentheorie auf die Koordinierung der Softwareentwicklung in dem untersuchten Open Source-Projekt KDE versucht. Die untersuchte Transaktion bezieht sich auf die organisationale, gemeinschaftliche Erstellung eines Softwareprodukts in dem

<sup>156</sup> Dies gilt für den Vergleich von Arbeitsergebnissen, aber auch für Angebote für spätere Arbeiten gleichermaßen. Software-Arbeitsergebnisse können schädigenden Quellcode beinhalten, der nicht durch die Wettbewerbsmerkmale erfasst wird. Bei Angeboten kann nicht ausgeschlossen werden, dass nicht schädigende Interessen vorhanden sind.

<sup>157</sup> Standortspezifität, Zeitspezifität und Humankapitalspezifität spielen dabei kaum eine Rolle, da das beste und passendste Arbeitspaket ausgewählt wird.

onale, gemeinschaftliche Erstellung eines Softwareprodukts in dem Open Source-Projekt KDE, wobei nach dem Eintritt von potenziellen Mitarbeitern eine Arbeitsplanung mit Arbeitsverteilung durchgeführt, ein individuelles Arbeitspaket erstellt und dann ihre Arbeit zusammengeführt wird (s.a. Kap. 2). Dabei beginnt die Transaktion als Arbeitsprozess in dem Teilprozess Eintritt/Rekrutierung mit dem ersten Kontakt. Danach erfolgt der Einstieg bzw. Mitgliedschaftsbeitritt in eine bestehende Softwareentwicklung beim Open Source-Projekt KDE. Die Transaktion endet mit der Beendigung der Softwareentwicklung eines offiziellen, stabilen Produkts.

In einem ersten Unterkapitel stehen die initialen Transaktionsprobleme mit ihren initialen Transaktionskosten in dem untersuchten Open Source-Projekt KDE im Mittelpunkt. Ihre Darstellung soll für ein besseres Verständnis des später dargestellten Einflusses der Koordinationsformen sorgen. Das KDE-Projekt basiert wegen seiner hohen Virtualität stark auf einer IuK-technischen Infrastruktur, weswegen der Einfluss der IuK-Technik mitbetrachtet wird. Einige initiale Transaktionsprobleme, die in Kapitel 6.1.1.3. und 6.1.1.4. größtenteils vorgestellt wurden, werden nun in den Teilprozessen der Softwareentwicklung (Eintritt/Rekrutierung; Arbeitsplanung mit Zieldefinition, detaillierter Entwicklungsplanung, Arbeitsverteilung; Leistungserstellung mit detaillierter Entwicklungsplanung; Arbeitszusammenführung) des Open Source-Projekts verortet. Da die detaillierte Entwicklungsplanung bei der Arbeitsverteilung in der Arbeitsplanung und bei der Leistungserstellung auftritt, sich aber inhaltlich als Planungsteilprozess kaum von der Zieldefinition unterscheidet, werden die initialen Transaktionsprobleme der detaillierten Entwicklungsplanung mit der Zieldefinition zusammen dargestellt. Bei der Analyse der Koordinationsformen in den Teilprozessen wird die detaillierte Entwicklungsplanung bei Arbeitsverteilung von Arbeitspaketen und Leistungserstellung untersucht.

In nachfolgenden Unterkapiteln wird der Einfluss der Koordinationsformen auf diese Transaktionsprobleme dargestellt, um die Bedeutung der Koordinationsformen in den Teilprozessen der Softwareentwicklung des Open Source-Projekts zu analysieren. Ausgangspunkt der Betrachtung ist der relative Unterschied zwischen den Koordinationsformen. Dabei wird angenommen, dass die initialen Transaktionskosten aller initialen Transaktionsprobleme konstant oder gleich hoch sind bzw. sich nicht unterscheiden, bevor Koordinationsformen diese beeinflussen. Im Zentrum steht die unterschiedliche Einflussstärke der Koordinationsformen auf die initialen Transaktionsprobleme. In den Teilprozessen der Softwareentwicklung des Open Source-Projekts wurden bei der empirischen Untersuchung Kombinationen der unterschiedlich bedeutenden Koordinationsformen vorgefunden, die als besonders effiziente Kombination der Koordinationsformen in einer Organisationsform bestehen blieben bzw. „überlebt“ haben, d.h. die geringsten Transaktionskosten gegenüber anderen Kombinationen hatten. Um die effizienteste Kombination der Koordinationsformen herauszufinden, wird der unterschiedliche Einfluss der drei Koordinationsformen auf die vorher angenommenen initialen Transaktionsprobleme verglichen. Die jeweilige Koordinationsform kann dabei negative und positive bzw. senkende und erhöhende Einflüsse auf die Transaktionskosten haben. Diese Einflüsse können auch über die genutzte IuK-technische Infrastruktur erfolgen, d.h. die IuK-technischen Einflüsse, die schon bei den initialen Transaktionsproblemen diskutiert wurden, können auch bei den Koordinationsformen eine Rolle spielen (s. Kap. 5.5.). Damit sind argumentative Überschneidungen möglich. Wenn eine Koordinationsform in einem Teilprozess zu hohe Transaktionskosten beinhaltet und somit an ihre Grenze stößt, mildert eine andere Koordinationsform diese Kosten an dieser Stelle. Je mehr Transaktionsprobleme in den Teilprozessen von einer Koordinationsform gemildert werden, desto bedeutender ist diese Koordinationsform.

### 6.1.2.1. *Initiale Transaktionsprobleme in den Teilprozessen*

In diesem Unterkapitel werden für das Open Source-Projekt KDE die auftretenden initialen Transaktionsprobleme dargestellt. Dabei wird beschrieben, wie diese initialen Transaktionsprobleme in dem Open Source-Projekt als Organisationsform auftreten können, d.h. welche initialen Transaktionsprobleme bei zusammenarbeitenden, softwareentwickelnden, autonomen, individuellen Akteuren bestehen können.

Wie oben beschrieben (s. Kap. 6.1.1.), treten manche initialen Transaktionsprobleme vor, manche nach einem Vertragsabschluss auf. Die initialen Transaktionsprobleme können daher den Teilprozessen der Softwareentwicklung zugeordnet werden, die sich auch in Beziehung zum Vertragsabschluss setzen lassen. Dabei treten Bereitstellungs- und Suchprobleme bei Eintritt/Rekrutierung sowie Zieldefinition und Arbeitsverteilung der Arbeitsplanung auf, d.h. in diesen Teilprozessen sind Probleme der Bereitstellung von Personen/Mitarbeitern oder Daten mit Informationsparadoxa, aber auch der Adverse Selection oder Information Overflow vorhanden. Bereitstellungs- und Suchprobleme wirken an diesen Stellen, da Eintritt/Rekrutierung mit einer externen Allokation von potenziellen Mitarbeitern und Arbeitsplanung mit einer internen Allokation von Mitarbeitern und Arbeitsaufgaben zusammenhängt. Da die Arbeitsplanung innerhalb des Open Source-Projekts KDE abläuft, können zudem auch Mess- und Spezifitätsprobleme auftreten. Nur Mess- und Spezifitätsprobleme bestehen bei der Leistungserstellung und Arbeitszusammenführung, wobei die konkrete Arbeit (Transformation der Arbeitskraft in Arbeitsleistung, s. Kap. 2) in Form der Softwareentwicklung erfolgt. Dabei sind die initialen Transaktionsprobleme bei Fringe Benefits/Free Rider-Verhalten, Shirking, Hold Ups und beim Nachverhandeln unvollständiger Verträge anzutreffen. Fringe Benefits und Free Rider-Verhalten erscheinen wie oben beschrieben beim Open Source-Projekt KDE als ähnliches Problem, da bei beiden der individuelle Nutzen auf Kosten der anderen Beteiligten erhöht wird. Im Weiteren wird aber eher auf das Free Rider-Verhalten rekurriert, da das Open Source-Projekt mit prinzipiell Gleichgestellten ein (quasi-)öffentliches Gut produziert.

Die Teilprozesse der Softwareentwicklung werden im Folgenden nacheinander auf die auftretenden initialen Transaktionsprobleme in dem Open Source-Projekt KDE untersucht.

#### 1. Eintritt/Rekrutierung

- Bei Eintritt/Rekrutierung besteht ein *Bereitstellungsproblem* auf Seiten der potenziellen Mitarbeiter dahingehend, ob genügend potenzielle Mitarbeiter zur Verfügung stehen und diese benötigtes, softwareentwicklungsspezifisches Humankapital und nicht nur allgemeine IuK-Technik-Kompetenzen mitbringen.<sup>158</sup> Auch wenn sich die Open Source-Bewegung augenscheinlich in den letzten Jahren stark vergrößert hat, interessieren sich nicht alle potenziellen Mitarbeiter für das Open Source-Projekt KDE. Besonders außerhalb dieser Open Source-Bewegung, aber auch innerhalb, entscheiden sich die potenziellen Mitarbeiter zwischen einem Einkommen für ihre Arbeit oder unentgeltlicher Arbeit in der Freizeit. Für das Open Source-Projekt KDE, das eher auf freiwilliger, unentgeltlicher Arbeit basiert, müssen daher vor allem die intrinsisch Motivierten angesprochen werden. Wenn aber potenzielle Mitarbeiter aus der Open Source-Bewegung ins Open Source-Projekt eintreten, wird im Allgemeinen davon ausgegangen, dass Adverse Selection-Probleme kaum auftreten (vgl. Benkler 2002). Da der Kontakt zwischen potenziellen und realen Mitarbeitern des Open Source-Projekts KDE von den potenziellen ausgeht und überwiegend über die

---

<sup>158</sup> Die geringere Standortspezifität durch die IuK-Technik lässt zwar einfacher Gleichinteressierte zusammenkommen (vgl. Döring 2003), doch müssen diese zueinander finden, was durch einen Information Overflow im Internet erschwert wird.

IuK-Technik erfolgt, fungiert eine Homepage als erster Anlaufpunkt für diese. Dabei kann das Bereitstellungsproblem etwas gemildert werden, wobei die KDE-Homepage nicht nur für die Selbstdarstellung, sondern auch für die projektinterne Kommunikation und Information verwendet wird. Durch die multiple Verwendung durch viele Personen werden über den Netzwerkeffekt<sup>159</sup> kaum zusätzliche Kosten verursacht.<sup>160</sup> Ein Informationsparadoxon hinsichtlich des vorhandenen Humankapitals eines potenziellen Mitarbeiters besteht weiterhin, da Humankapital nicht vollständig digitalisierbar ist. Potenzielle Mitarbeiter können sich wenigstens projekt- und softwareentwicklungsspezifisches Humankapital durch den offenen Zugriff auf die Infrastruktur des Open Source-Projekts aneignen und so das Bereitstellungsproblem hinsichtlich des Humankapitals etwas entschärfen. Probleme, dass Daten und Informationen nicht digitalisiert und übermittelt werden können, führen u.a. zu Missverständnissen und einem verringerten Interesse an dem Projekt.

- Sind überhaupt potenzielle und direkt einsetzbare Mitarbeiter vorhanden, müssen diese noch gesucht werden. Das *Suchproblem* hängt vor allem mit Adverse Selection zusammen, da nicht genügend relevante Informationen über potenzielle Mitarbeiter über das Internet herangezogen werden können. Gerade durch die ansteigenden Nutzungsraten des Internets mit zunehmender Internetaktivität (s.o.) besteht ein Information Overflow. Es wird immer aufwändiger, die eher wenigen potenziellen Mitarbeiter mit entsprechendem Humankapital und intrinsischer Motivation für einen direkten Arbeitseinsatz bei der Softwareentwicklung in einem Open Source-Projekt zu suchen und anzusprechen. Außerdem sind möglicherweise opportunistische potenzielle Mitarbeiter auszusortieren, die schädigenden Quellcode schreiben wollen bzw. könnten oder den Quellcode außerhalb des Lizenzvertrages in anderer Software verwenden könnten. Ein zusätzliches Hindernis sind irrelevante oder falsche Informationen. Dieses Problem ist generell vorhanden, da ein Anonymitätsproblem im Internet bzgl. falscher Identität und unüberprüfbarer Angaben besteht und eine asymmetrische Informationsverteilung fördert. Wird Software von externen potenziellen Mitarbeitern eingereicht, hindert der Erfahrungsgut-Charakter der Software eine vollständige Bewertung, so dass schädigender Quellcode z.B. von Viren eingeschmuggelt werden kann.

Generell gilt für alle nachgelagerten Teilprozesse, dass sich die Wahrscheinlichkeit eines Hold Ups der bisherigen Mitarbeiter mit hohem Wissen gegenüber Mitarbeitern mit wenig Wissen erhöht, wenn nicht weiterhin genügend Mitarbeiter vorhanden sind bzw. eintreten/geworben werden. Dies gilt auch für das Open Source-Projekt KDE, das zwar aus ca. 1000 Mitarbeitern besteht, die sich aber auf verschiedene Bereiche verteilen.

## 2. Arbeitsplanung

Der Teilprozess Arbeitsplanung zerfällt in Zieldefinition, detaillierte Entwicklungsplanung und Arbeitsverteilung, in denen entweder Alternativen/Ideen/Ziele, Arbeitsaufgaben oder verantwortliche Posten kommuniziert, diskutiert und verteilt werden. In dem Open Source-Projekt KDE beeinflusst der Teilprozess Eintritt die nachfolgenden Teilprozesse, so dass die Wahrscheinlichkeit für ein Auftreten eines initialen Transaktionsproblems in einem anderen Teilprozess niedrig sein kann.

An dieser Stelle wird zuerst die detaillierte Entwicklungsplanung gemeinsam mit der Zieldefinition diskutiert, da beide Unterteilprozesse sich um die Entscheidung für ein Ziel/eine

---

<sup>159</sup> Netzwerkeffekt heißt, dass der Nutzen eines Guts für die Akteure mit der Anzahl der Akteure ansteigt, die diesen verwenden. Dabei sind die Produktionskosten relativ hoch, wobei die Distributionskosten pro Abnehmer konstant bleiben (Arentzen, Lörcher 1997).

<sup>160</sup> Auf Linux-Konferenzen ist dies ähnlich. Das Projekt stellt sich dabei mit einem Stand vor und dient als Anlaufpunkt u.a. auch für eine Rekrutierung von Interessierten.

Alternative mit verschiedenen Detaillierungsgraden drehen. Danach wird auf die Arbeitsverteilung von Aufgabenbereichen/-paketen und durch Postenbesetzung eingegangen, wobei es sich entweder um temporäre oder unbefristete Aufgaben bzw. um Aufgaben mit unterschiedlichem Verantwortungsumfang handelt.

Neben Bereitstellungs- oder Suchproblemen treten u.a. auch Mess- und Spezifitätsprobleme auf (s.o.). Shirking tritt nicht auf, da an dieser Stelle noch keine Arbeitsaufgaben durchgeführt oder erstellt werden. Auch Nachverhandlungen treten erst auf, nachdem die Mitarbeiter Arbeitsaufgaben erledigt haben und dabei in die Beziehung zum Projekt investiert haben.

## 2.1. Arbeitsplanung: Zieldefinition und detaillierte Entwicklungsplanung

Während in der wirtschaftlichen Sphäre normalerweise die Wünsche des Kunden bzw. Endnutzers die Zieldefinition und detaillierte Entwicklungsplanung überwiegend bestimmen, geschieht dies beim Open Source-Projekt KDE durch die Softwareentwickler selbst über Ziele und die detaillierte Entwicklungsplanung. Da die detaillierte Entwicklungsplanung beim Open Source-Projekt KDE mit der Leistungserstellung und Arbeitsverteilung zusammenhängt und oftmals von den Softwareentwicklern selbst durchgeführt wird, bestehen dabei kaum initiale Transaktionsprobleme. Im Weiteren werden mögliche initiale Transaktionsprobleme vorgestellt, die bei der Zieldefinition, aber auch bei der detaillierten Entwicklungsplanung auftreten können, wenn die Softwareentwickler sich koordinieren.

- Die Projektmitarbeiter generieren Alternativen/Ideen/Ziele, die mit *Bereitstellungs- oder Suchproblemen* zusammenhängen.<sup>161</sup> Entweder werden schon bereitstehende, passende Ideen aus anderen Open Source-Projekten übernommen oder neue Lösungen erdacht. Die Kosten für die Übernahme bisher bereitstehender, bekannter Ideen aus anderen Open Source-Projekten umfassen hohe Suchkosten nach einer passenden Lösung. Wenn aber keine bekannten Ideen bereitstehen, wird eine eigene Lösung gesucht, die höhere Transaktionskosten beinhaltet. Zuerst werden Ziele und Ideen vorgeschlagen, dann Auswahlkriterien festgelegt und schließlich eine glaubwürdige Lösung ausgewählt. Mit einem Information Overflow ist zu rechnen, da viele Ideen von vielen Mitarbeitern im Open Source-Projekt KDE und auch von Projektexternen in den Mailinglisten vorgeschlagen werden können. Dabei kann ein Information Overflow verstärkt auftreten, je mehr ein Mitarbeiter in das Open Source-Projekt involviert ist und viele Mailinglisten abonniert hat. Auftretende Information Overflows werden aber durch die Aufteilung in verschiedene themenbezogene Mailinglisten abgeschwächt, wobei die Information Overflows stärker abgeschwächt werden, je konkreter und spezialisierter das Diskussionsthema ist. Dabei wird die detaillierte Entwicklungsplanung tendenziell eher in den spezialisierten Mailinglisten und allgemeine projektumfassende Ziele in einer projekt- und themenübergreifenden zentralen Mailingliste mit dem höchsten Information Overflow diskutiert.

Aber je komplexer eine Idee ist, desto schwieriger und kostenintensiver ist diese über die luK-Technik zu kommunizieren. Dabei kann ein einfacheres, weniger arbeitsintensiv erscheinendes Ziel/eine Aufgabe ausgewählt werden, obwohl eine andere Alternative, die zwar anfänglich komplizierter, arbeitsintensiver erschien, eher zum Erfolg geführt hätte (Adverse Selection). Bei einer späteren Umorientierung führt dies wiederum zu zusätzlichen Suchkosten, verringert jedoch die Kosten der Leistungserstellung/detaillierten Entwicklungsplanung wie bei der Programmiersprachenumstellung. Wird an bisher Program-

---

<sup>161</sup> Weitere Probleme wie ein Informationsparadoxon spielen keine Rolle, da die bekannte Idee bzw. Quellcode zugänglich ist. Außerdem spielen Patent- oder Lizenzgebühren bei Open Source-Projekten bisher keine Rolle.

miertes angeknüpft, dann wird die Auswahl an Alternativen bzw. Ideen der Softwareprogrammierung durch den bisherigen Quellcode eingeschränkt, was die Kosten begrenzt.<sup>162</sup> Dieses Problem wird durch eine zusätzliche Anbindung der detaillierten Entwicklungsplanung an die Leistungserstellung vermindert, da durch die zeit- und problemnahe Lösungssuche die Kosten geringer sind.

- Das *Messproblem* Free Rider-Verhalten kann auftreten, wenn die Mitarbeiter Ziele oder Aufgaben besonders diskutieren, die die höchste spätere Bewunderung durch Freunde oder Endnutzer bringen, hierdurch aber andere Aufgaben vernachlässigen. Oder Mitarbeiter wählen aus mangelndem Humankapital oder der Erwartung von hohem Arbeitsaufwand eine Alternative, die den geringsten Arbeitsaufwand verspricht, das Projekt jedoch auf lange Sicht nicht voranbringt bzw. höhere Kosten verursacht. Ein oder mehrere Mitarbeiter können eine ihnen genehme Softwareentwicklung anstreben, um die Softwareentwicklung nach ihren z.B. wirtschaftlichen Interessen entgegen der Interessen des Open Source-Projekts auszurichten. Im Open Source-Projekt KDE haben schon Unternehmen versucht, durch eine geschickte Platzierung ihrer Angestellten im Projekt, die Entwicklungsrichtung im Sinne ihrer Unternehmensinteressen entgegen der KDE-Projekt-Interessen zu beeinflussen. Dabei geht es nicht nur um die Verhinderung von konkurrierenden Softwareentwicklungen oder die Anpassung an eigene proprietäre Software, sondern auch um die mögliche unlicenzierte Zweitverwendung des Quellcodes in proprietären Eigensoftwareentwicklungen z.B. Shareware. Diese Problematik kann durch eine falsche Identität verstärkt werden.

- Einzelne Mitarbeiter können diese Messprobleme durch *Spezifitätsprobleme* verstärken, indem sie eine Machtstellung verwenden, um ihnen Nutzenbringendes durchzusetzen.<sup>163</sup> Dies kann als Hold Up bezeichnet werden, da beim Open Source-Projekt KDE der individuelle Arbeitsaufwand nicht nur eine Investition des einzelnen Mitarbeiters, sondern auch eine Investition für das ganze Projekt darstellt. Folglich kann eine Person über eine gegenseitige Abhängigkeit eine Person einer anderen ihren Willen aufzwingen. Ein Mitarbeiter hat sich projektspezifisches Humankapital angeeignet, das dem Projekt neben seiner Arbeitskraft fehlen würde. Die Opportunitätskosten eines Ersatzes wären dabei hoch. Eine Aufkündigung der Zusammenarbeit mit bestimmten Mitarbeitern kann auch als Druckmittel eingesetzt werden. Ein weiteres Druckmittel wäre, wenn ein Mitarbeiter mit hohem projektspezifischen Humankapital ständig für eine Lösung argumentiert, um diese für ihn besonders interessante Lösung durchzusetzen. Ein Hold Up hinsichtlich der Reputation entsteht, wenn ein Mitarbeiter droht, durch provozierendes und nicht-normenkonformes Verhalten oder den Einbau von beabsichtigten Fehlern oder Computerviren in die Software die Reputation des Projekts zu zerstören, da sein Verhalten von den Projektexternen auf das Projekt verallgemeinert werden kann (Markennamenspezifität). Mit dieser Drohung beabsichtigt der Mitarbeiter, eine bevorzugte Alternative durchzusetzen.<sup>164</sup>

---

<sup>162</sup> Wird entschieden, große Teile bisher geschriebenen Quellcodes zu löschen und neuen zu schreiben, dann erfordert dieser seltene Fall generell höhere Kosten bei der detaillierten Entwicklungsplanung und der Leistungserstellung und u.U. bei der Arbeitszusammenführung. Dies gilt für alle Koordinationsformen gleichermaßen und geht daher mit der Begründung für die übliche Vorgehensweise konform.

<sup>163</sup> Projektexterne haben diese Machtstellung i.W. nicht.

<sup>164</sup> Hold Ups aus anlage- oder abnehmerspezifischen Investitionen spielen in einem Open Source-Projekt keine Rolle. Die Privatpersonen investieren kaum in Anlagen für die Arbeit im Projekt, sondern nutzen die eigene, vorhandene Hardware und die frei verfügbare Software des Projekts. Da die Softwareentwickler zugleich auch Nutzer der Software sind (s. Kap. 2), bestehen keine abnehmerspezifischen Investitionen.

## 2.2. Arbeitsplanung: Arbeitsverteilung von Arbeitspaketen und verantwortlichen Posten mit Aufgabenbereichen

Bei dem Teilprozess Arbeitsverteilung werden Arbeitspakete oder verantwortliche Posten mit Mitarbeitern zusammengeführt. Da sich Arbeitspakete und Posten von der Transaktionskostentheorie aus gesehen kaum unterscheiden, weil es sich prinzipiell um Aufgaben mit unterschiedlichem Verantwortungsbereich und Zeithorizont handelt, werden beide Teilbereiche hier zusammen als Aufgaben betrachtet.

- Durch den Eintritt/die Rekrutierung in das Open Source-Projekt KDE wird das *Bereitstellungsproblem* etwas gemildert. Glaubwürdige Mitarbeiter mit entsprechender IuK-technischer Kompetenz und softwareentwicklungsbezogenem Humankapital sind im Open Source-Projekt KDE in ausreichender Anzahl vorhanden und können Aufgaben übernehmen. Demgegenüber schränken Arbeitsauslastung bzw. multiple Anforderungen an Mitarbeiter eine Übernahme von Aufgaben ein, d.h. diese Mitarbeiter stehen nicht zur Verfügung. Eine Unsicherheit im Zusammenhang mit zukünftigen Umweltzuständen verhindert zudem die Übernahme von Aufgaben, da von vornherein nicht klar ist, wie umfangreich die Aufgaben sind. Manche Aufgaben sind auch schwierig zu kommunizieren und werden daher vernachlässigt. Gerade Posten werden nicht gern übernommen, weil damit ein hoher Arbeitsaufwand verbunden ist.

- Ein *Suchproblem* besteht bei der Arbeitsverteilung in abgeschwächter Form, da in dem Open Source-Projekt KDE die Arbeit freiwillig übernommen wird und keine Instanz Aufgaben zuweist. Das Suchproblem besteht jedoch weiterhin aufgrund fehlender und passender Mitarbeiter, so dass auch wichtige Arbeitsaufgaben nicht weiterverfolgt werden. Das Suchproblem wird besonders hinsichtlich des Humankapitals gemildert, da der überwiegende Anteil der Mitarbeiter eine Mindestqualifikation mitbringt und eine Aufgabe nach deren Wissensstand auswählt. Manche haben aber höheres projektspezifisches Humankapital als andere. Es wird eher eine Person gesucht, die eine Aufgabe nach Abschluss einer anderen bzw. zusätzlich dazu übernehmen könnte.<sup>165</sup> Bei über 1000 Mitarbeitern kann aber auch ein Information Overflow auftreten, so dass die Suchkosten ansteigen. Umgekehrt kann ein Mitarbeiter eine Aufgabe suchen, muss aber einen Information Overflow z.B. von Fehlerberichten und Programmierwünschen sowie irrelevante Informationen über Aufgaben aussortieren. Die verschiedenen themenbezogenen Mailinglisten oder das Fehlerberichtssystem schwächen das Suchproblem ab, da abhängig von Komplexität/Arbeitsumfang bzw. Themenbezug eine Aufgabe oder neue Mitarbeiter gesucht werden können. Sind jedoch nicht genügend ansprechbare Mailinglistennutzer oder Aufgaben in einem Aufgabengebiet vorhanden, wird zusätzlich über andere Mailinglisten gesucht, was aber wegen der Freiwilligkeit nicht unbedingt zum Erfolg führt. Ein Information Overflow kann weiterhin bestehen, weil über die meisten Mailinglisten oder das Fehlerberichtssystem auch Projektexterne Informationen kommunizieren können. Zusätzlich können Konflikte um die Übernahme von bestimmten Arbeitsaufgaben oder Posten Kommunikationskosten erhöhen, weil Aufgaben freiwillig übernommen werden. Auch wenn kleine Arbeitspakete parallel erstellt wurden und die Bearbeiter nichts voneinander wussten, erhöht dies die Kommunikationskosten.

- Auch hier zeigt sich Free Rider-Verhalten als *Messproblem*, wenn die Ziele oder die Aufgaben besonders ausgewählt werden, die die Bewunderung durch Freunde oder für die Endnutzer garantiert. Für das Projekt wichtigere Aufgaben, die für Projektexterne aber

---

<sup>165</sup> Da im Open Source-Projekt KDE über 1000 Mitarbeiter beteiligt sind, sind die Suchkosten höher als in einem Projekt mit fünf Personen, aber geringer als auf einem Arbeitsmarkt mit Tausenden von Bewerbern.

nicht ersichtlich sind, werden dadurch vernachlässigt. Ähnlich verhält es sich, wenn die Mitarbeiter Aufgaben auswählen und dabei ihr Humankapital, ihr Arbeitsvermögen und ihre Arbeitszeit überschätzen. Der Mitarbeiter braucht länger, eine bestimmte Aufgabe zu lösen oder zu erfüllen, wäre aber bei einer anderen einfacheren Aufgabe schneller am Ziel und könnte eine neue Aufgabe übernehmen. Mitarbeiter können aber auch Aufgaben absichtlich umgehen, bei denen von Anfang an klar ist, dass diese einen hohen Arbeitsaufwand umfassen. Mitarbeiter hoffen dann, dass ein anderer Mitarbeiter diese Aufgabe übernimmt und seine Zeit opfert. Auch kann ein Mitarbeiter oder eine Person mit falscher Identität eine Aufgabe wählen, um den später geschriebenen Quellcode proprietär zu verwenden (vgl. Benkler 2002).

- *Spezifitätsprobleme* in Form von Hold Ups treten an dieser Stelle auch eher selten auf, da die Übernahme einer Aufgabe freiwillig ist (vgl. Benkler 2002). Es können aber Mitarbeiter mit einer Machtstellung und hohem projektspezifischem Humankapital andere ohne Machtstellung und hohes projektspezifisches Humankapital von Aufgaben abbringen und diese selbst übernehmen. Weiterhin kann über einen Hold Up hinsichtlich der projektbezogenen Reputation (s.o.) ein Mitarbeiter die Übernahme einer Aufgabe erzwingen (Markennamenspezifität).

### 3. Leistungserstellung und Arbeitszusammenführung

Auch die Teilprozesse der Leistungserstellung und Arbeitszusammenführung ähneln sich stark, da in beiden Fällen konkret Software als vorläufiger Entwicklungsrelease oder als offizieller Release erstellt wird. Der Unterschied, dass bei der Arbeitszusammenführung ein Endtermin mit einem offiziellen, fehlerfreien Softwareprodukt angestrebt wird, fällt an dieser Stelle nicht ins Gewicht, verschärft aber manche Transaktionsprobleme. Beide Teilprozesse werden daher zusammen betrachtet.

- Bei der Softwareentwicklung kann ein Free Rider-Verhalten als *Messproblem* auftreten, wenn eigene, projektschädigende Ziele verfolgt werden. Beispiele für eigene, projektschädigende Ziele sind, wenn trojanische Spionageprogramme der Computerfernsteuerung/-schädigung, beabsichtigte Fehler oder Computerviren in den Quellcode eingebaut werden. Oder wenn aus anderen Gründen, etwas anderes programmiert wird als angekündigt. Dabei können auch Teile von Quellcode gelöscht werden, um andere zu ärgern. Der Erfahrungsgut-Charakter der Software, mögliche falsche Identitäten, aber auch die Information Overflows durch häufige Änderungen der vielen Open Source-Mitarbeiter unterstützen dabei Free Rider-Verhalten. Im Projekt werden diese Probleme jedoch abgemildert, da über Versionsmanagementsysteme die Wiederherstellung von Quellcode ohne Weiteres möglich ist und dessen Modularität eine aufwandsarme Feststellung von schädigendem Verhalten ermöglicht (vgl. Brand, Holtgrewe 2007). Der geschriebene Quellcode könnte auch einer nicht-lizenzierten Zweitverwendung zugeführt werden, d.h. das gesamte Produkt oder Teile des erstellten Quellcode werden in anderer, proprietärer Software verwendet und dafür Lizenzgebühren verlangt. Quellcode kann auch zuerst im Projekt geschrieben, dann gelöscht und mitgenommen werden. Dies wird auch durch die im Internet mögliche Verwendung von falschen Identitäten besonders ermöglicht. Information Overflows auf den Mailinglisten treten besonders bei der Arbeitszusammenführung auf, da mehrere Diskurse z.T. parallel verlaufen und sich aufeinander beziehen können.

Zusätzlich besteht eine mögliche Shirking-Problematik nur begrenzt, da die Mitarbeiter Aufgaben nach ihrer Motivation auswählen. Diese Motivation kann sich aber schnell ändern, so dass Mitarbeiter einen Arbeitsbereich übernehmen können. Andere Mitarbeiter sind aber von der Arbeit abhängig, wobei die Abhängigkeit mit der Zentralität der Software und dem projektspezifischen Humankapital des Mitarbeiters für das Projekt ansteigt. Dies

führt wiederum zu Kompetenz-, Ablöse- und Verantwortlichkeitsstreitigkeiten. Weiterhin ist bei der Zusammenarbeit in dem Open Source-Projekt KDE eine Unterscheidung einer besten von einer minimalen Leistung kaum möglich. Das, was ein Mitarbeiter berichtet, ist wegen der Virtualität z.T. schwierig nachzuvollziehen (vgl. Alchian, Demsetz 1972; Jones et al. 1997). Bei der Kommunikation ist auch eine Unterscheidung zwischen ungewollten Missverständnissen und Opportunismus ebenfalls schwierig. Dabei nimmt der Opportunismus die Form von nicht eingehaltenen Absprachen, um Arbeit zu vermeiden, aber auch von (falschen) Anschuldigungen, Rufmord oder Verleumdungen an. Zusätzlich berufen sich die Mitarbeiter zuweilen auch auf Missverständnisse, um ihr opportunistisches Verhalten zu kaschieren. Zudem sind auch Information Overflows der Mitteilungen, aber auch der Anschuldigungen möglich, da viele Open Source-Mitarbeiter, aber auch Projektexterne über die Mailinglisten kommunizieren.

- Nach Eintritt wird in dem Open Source-Projekt KDE von allen Mitarbeitern Arbeitszeit investiert und projektspezifisches Humankapital aufgebaut, so dass wie oben beschrieben eine Investition für das ganze Projekt durch die gegenseitige Abhängigkeit für einen Hold Up als *Spezifitätsproblem* missbraucht werden kann. Vor wichtigen Terminen wie einem offiziellen Release können ein oder mehrere Mitarbeiter die Übernahme von bestimmten Programmfunktionen oder Softwareteilen erzwingen, vor allem wenn es sich um projektzentrale oder komplexe Software handelt. Mitarbeiter mit hohem projektspezifischem Humankapital können den Quellcode von anderen mit geringerem Humankapital kritisieren, ausschließen und durch eigenen ersetzen. Ob dies gerechtfertigt ist, kann bei hoch komplexer Software schwer festgestellt werden. Auch können ein oder mehrere Mitarbeiter mit hohem projektspezifischem Humankapital die Software kidnappen bzw. übernehmen und von einer freien zu einer proprietären Lizenz ändern (Benkler 2002: 36).

#### *6.1.2.2. Analyse der Koordinationsformen in den Teilprozessen*

Im vorigen Unterkapitel wurden die initialen Transaktionsprobleme in den Teilprozessen der Softwareentwicklung untersucht. In den nachfolgenden Unterkapiteln steht der Einfluss der Koordinationsformen in den Teilprozessen der Softwareentwicklung im Fokus. Dabei wird der Stellenwert der vorgefundenen Koordinationsformen analysiert, wobei es sich jeweils die gleiche Software handelt. Auch in diesem Unterkapitel werden wie bereits im vorangegangenen Vermutungen aufgestellt.

Bei dem Vergleich wird die Kombination der Koordinationsformen mit der höchsten Effizienz gesucht. Die Effizienz beschreibt das Verhältnis zwischen maximalem Output und kostenminimalem Input bei den Koordinationsformen, das durch einen Vergleich der Gesamttransaktionskosten unter Einfluss einer Koordinationsform bestimmt wird. Dabei sind die initialen Transaktionskosten bei den Koordinationsformen konstant und gleich hoch, da es sich um die gleiche Transaktion bei derselben Software handelt. Der relative Unterschied zwischen den Koordinationsformen wird untersucht, d.h. das Auftreten der initialen Transaktionsprobleme wird unabhängig vom Einfluss der Koordinationsformen gesehen. Verglichen wird damit das Verhältnis Output zum Input der Koordinationsformen, wobei der Output den Gesamteinfluss auf die Transaktionskosten beinhaltet und der Input die Instrumentnutzungskosten. Dabei ist diejenige Koordinationsform am effizientesten, die den größten Gesamteinfluss auf die Transaktionskosten bei den niedrigsten Instrumentnutzungskosten hat, d.h. die bei konstanten Produktionskosten die geringsten Gesamttransaktionskosten beinhaltet. Damit sollte eine Koordinationsform, um effizient zu sein,

viele bzw. die meisten initialen Transaktionsaktionsprobleme mildern. Die, die sie nicht mildern kann, können durch andere Koordinationsformen abgeschwächt werden.<sup>166</sup>

Wegen der qualitativen, eher ordinalen Daten kann der Einfluss auf die initialen Transaktionsprobleme nur grob geschätzt werden. Da zusätzlich alle drei Koordinationsformen u.a. alle auftretenden initialen Transaktionsprobleme beeinflussen, wird ihre Bedeutung mittels einer Rangfolge anhand folgender Klassifikation festgestellt. Die Koordinationsformen können nur stark oder schwach Einfluss nehmen, d.h. die Gesamttransaktionskosten werden schwach oder stark verringert. Die geringsten Gesamttransaktionskosten mit der effizientesten Koordinationsform bestehen bei einer starken bis vollständigen Senkung der initialen Transaktionskosten und einer schwachen bzw. keiner Erhöhung bei den Nutzungskosten. Umgekehrt entstehen höhere Gesamttransaktionskosten bei einer schwachen Senkung der initialen Transaktionskosten und einer starken Erhöhung bei den Nutzungskosten. Mittlere Gesamttransaktionskosten bestehen, wenn entweder beide Transaktionskosten verringert oder erhöht werden (s. Tab. 6-1). In manchen Fällen kann zudem begründet werden, warum eine Koordinationsform nur etwas effizienter als eine andere ist, d.h. z.B. ein geringer relativer Abstand zwischen hoch effizienter und mittel effizienter Koordinationsform besteht. Generell gilt, dass mit höherer Effizienz einer Koordinationsform eine höhere Bedeutung dieser Koordinationsform einhergeht.

**Tab. 6-1:** Effizienz der jeweiligen Koordinationsform durch Einfluss auf die jeweiligen Transaktionskosten, eigene Darstellung

Einfluss auf Gesamttransaktionskosten		Erhöhung der Nutzungskosten der Koordinationsformen (durch Handlung)	
		stark	schwach
Senkung der initialen Transaktionskosten (durch Handlungsbedingung und Handlung)	stark	mittel	effizient
	schwach	ineffizient	mittel

Im Folgenden werden nun bei den Teilprozessen des Open Source-Projekts KDE (Eintritt/Rekrutierung; Arbeitsplanung: Zieldefinition, Arbeitsverteilung von Aufgabenbereichen/Arbeitspaketen mit detaillierter Entwicklungsplanung, Arbeitsverteilung durch Postenbesetzung; Leistungserstellung mit detaillierter Entwicklungsplanung und Arbeitszusammenführung) die Kombinationen der Koordinationsformen analysiert, wobei jeweils der

<sup>166</sup> Der Einfluss auf die Gesamttransaktionskosten kann folgendermaßen veranschaulicht werden: Durch einen Vergleich der Gesamttransaktionskosten beim Einfluss einer einzigen Koordinationsform, kann eine effiziente Kombination der Koordinationsformen festgestellt werden, was sich durch folgende Formel darstellen lässt.

$$TK_{\text{Gesamt, Koord.form 1}} > TK_{\text{Gesamt, Koord.form 2}} > TK_{\text{Gesamt, Koord.form 3}}$$

Dabei ist

$$TK_{\text{Gesamt, i}} = TK_{\text{initial, j}} - TK_{\text{Koord.form, i}} + TK_{\text{Koord.formnutzung, i}} > 0,$$

wobei Index i hier für Koordinationsform Kooperation, Hierarchie oder Wettbewerb steht. „- TK<sub>Koord.form</sub>“ stellt den Nutzen einer Koordinationsform „N<sub>Koord.form</sub>“ und „TK<sub>Koord.formnutzung i</sub>“ die Instrumentnutzungskosten dar. Außerdem entstehen die konstanten „TK<sub>initial, j</sub>“ aus der Summe der Bereitstellung-, Such-, Mess- und Spezifitätsprobleme als Index j.

Einfluss auf die initialen Transaktionsprobleme betrachtet wird. Die Teilprozesse werden nacheinander in je einem Unterkapitel untersucht, wobei die Koordinationsformen nach ihrer absteigenden Bedeutung aufgeführt werden.

Generell tritt strategische Kooperation im Open Source-Projekt nicht auf, da vermutlich wegen der Informalität des Projekts institutionenbasiertes Vertrauen nicht auftritt. Auch wenn ein Verein für die Außendarstellung, Kanalisierung der Gelder etc. gegründet wurde, hat er keinen Einfluss auf die hier untersuchte Softwareentwicklung. Auch kann vorheriges kalkulatives Vertrauen eine strategische Kooperation nicht unterstützen, da sich aufgrund der Größe des KDE-Projekts nicht alle Personen vorab genauer kennen lernen können. Außerdem bestehen keine nutzenbringenden Güter, die getauscht werden können. Quellcode ist generell wegen der GPL-Lizenz offen zugänglich und eine Entlohnung spielt wegen der vorherrschenden intrinsischen Motivation keine Rolle.

### 6.1.2.3. Eintritt/Rekrutierung

Beim Eintritt als direkter Kontakt zwischen Open Source-Projekt KDE und potenziellen Mitarbeitern können die Koordinationsformen das Bereitstellungsproblem von potenziellen Mitarbeitern mit spezifischem Humankapital und vorwiegend spaßbasierter, intrinsischer Motivation ohne Zwänge nicht lösen, aber zumindest auch nicht verstärken. Hierarchie/Macht und Wettbewerb funktionieren über extrinsische Motivationsgründe wie materielle Anreize oder Strafen bzw. Ausschreibungen, die von außen vorgegeben werden. Sie zielen nicht auf intrinsisch Motivierte z.B. aus der sozialen Open Source-Bewegung, die angesprochen werden sollen. Materielle Anreize sind durch die Virtualität mit hohen Kosten z.B. des Versendens verbunden. Finanzielle Anreize als Spenden haben eher einen Mitnahmeeffekt (Hobby ist Beruf), als dass sie die Motivation weiter steigern könnten. Strafen sind wegen fehlender Abhängigkeiten der potenziellen Mitarbeiter auch nicht denkbar. Ausschreibungen treten wegen fehlender verknappter Mitgliedschaften/Stellen oder Anreize ebenso wenig auf. Kooperatives Verhalten mit freundlicher Kommunikation behindert bei geringen Kommunikationskosten auch die freiwillige und intrinsische Motivation nicht, wirkt aber wie eine Einladung zu einem Austausch über das Angestrebte und unterstützt durch ein minimales Vertrauen das freiwillige Engagement.<sup>167</sup> Unterschiedliche Einflüsse zeigen sich aber bzgl. der Suchprobleme.

- Beim Eintritt ist *Hierarchie/Macht* effizient und damit hoch bedeutend, da neben einer hohen Senkung der initialen Transaktionskosten kaum Instrumentnutzungskosten auftreten. Das Prozedere des Einstiegs wird auf der Homepage als Willensbekundung dargestellt. Die potenziellen Mitarbeiter müssen sich an die Regeln halten bzw. minimale Standards einhalten, sonst bekommen sie keinen eigenen Zugang zum Versionsmanagementsystem und damit zum Projekt. Die Anstrengungen der potenziellen Mitarbeiter zählen dabei nicht zur Transaktion. Der Zutritt stellt eine Belohnung dar, die verdient werden muss. Eine weitere Anreizsetzung ist wegen der intrinsischen Motivation nicht nötig und wirkt u.U. abschreckend.

Niedrige Instrumentnutzungskosten treten erstens auf, da das Prozedere des Einstiegs über die Homepage verbreitet wird, wobei über einen Netzwerkeffekt (s.o.) geringe Kosten verursacht werden. Zweitens bestehen kaum Kommunikationskosten durch das transaktionskostenminimale Ignorieren von irrelevanten Informationen auf der Mailingliste und von potenziellen Mitarbeitern mit geringem projektspezifischem Humankapital. Drittens müssen die potenziellen Mitarbeiter, um Aufmerksamkeit zu erhalten, über ihre Arbeit berich-

---

<sup>167</sup> Dieses erste Grundvertrauen lässt weiteres kooperatives Verhalten erst zu (vgl. Tit-for-Tat-Strategie; Holler, Illing 1996).

ten bzw. Quellcode einreichen (Reporting). Dadurch wird auch ein Informationsparadoxon gemildert. Viertens treten kaum Kontrollkosten auf, da im Wesentlichen kurze Programmteile von wenigen Posten beurteilt werden. Da jeweils ein Projektleiter für einen potenziellen Eintretenden vorhanden ist, bestehen geringe Entscheidungskosten. Werden längere Programme aufgenommen, werden diese nur von wenigen KDE-Mitgliedern getestet, was auch geringe Kontroll- und Entscheidungskosten nach sich zieht.

- *Wettbewerb* als mittel effiziente bzw. bedeutende Koordinationsform senkt kaum die initialen Transaktionskosten z.B. des Informationsparadoxons, da sich die potenziellen Mitarbeiter nicht gegenseitig erkennen und beobachten können, da der Projektleiter den eingegebenen Quellcode ins Versionsmanagementsystem stellt. Es gibt zwar Mailinglisten, in denen potenzielle Mitarbeiter eher auftreten, doch sind diese durch Information Overflow schwer auszumachen, auch da sie sich seltener an der Kommunikation beteiligen. Die geringen Anpassungs- und Verbesserungskosten (s. selbstorganisierte Sozialisation) fallen beim potenziellen Mitarbeiter an bzw. werden vom Projekt auf diesen abgewälzt und zählen nicht zur Transaktion, auch da diese als Sunk Costs die Bereitschaft zur Beteiligung signalisieren.

Die Mitgliedschaft durch Schreibberechtigung ist als Organisationsformengrenze zwar ein relativ verknapptes Gut, doch ist eine Ausweitung der Mitgliedschaften ohne Weiteres möglich. Da die Anzahl der Mitgliedschaften kaum verknappt ist, ist mit geringen Senkungen der initialen Transaktionskosten zu rechnen, da sich die potenziellen Mitarbeiter nicht außerordentlich anstrengen. Dagegen ist die Aufmerksamkeit eines Postens wegen seiner stark begrenzten Arbeitszeit und einem Information Overflow knapper. Diese Aufmerksamkeit kann nur durch funktionierenden Quellcode, den der Projektleiter kontrolliert (s.o.) und ins Versionsmanagementsystem einbaut, und hoch qualifizierte Fragen erworben werden. Dabei vergleicht sich der potenzielle Mitarbeiter mit den aktuellen KDE-Mitarbeitern, was mangels projektspezifischen Humankapitals aber wiederum initiale Transaktionskosten nur wenig senkt. Der vorhandene Information Overflow wird zudem etwas gemildert, da sich nur potenzielle Mitarbeiter mit passendem Humankapital bei den Subprojekten beteiligen (Self Selection der Akteure), deren Software sie interessiert.

Die Nutzungskosten des Wettbewerbs steigen beim Eintritt auch nur geringfügig, da nur manchmal der Projektleiter den eingereichten Quellcode mit anderem vergleicht. Zudem ist Wettbewerb zwischen den Open Source-Projekten um potenzielle Mitarbeiter gering. Es wird zwar über einen Internetauftritt, Suchanzeigen auf Mailinglisten/Newsplattformen, Zeitschriftenartikel etc. Werbung als Signalling betrieben, wobei die Marketingkosten und Erstellungskosten wegen des Netzwerkeffekts und einer geringen Anzahl von Bearbeitern gering gehalten werden. Außerdem gehören die Projekte der gleichen sozialen Open Source-Bewegung an, wobei eine direkte Abwerbung bzw. Bewerbung auf informaler Ebene nicht gern gesehen wird.

- *Kooperation* ist dagegen ineffizient und damit wenig bedeutend, da das Signalisieren gemeinsamer Ziele, das mit Vertrauen verbunden ist, bei hoher Virtualität schwierig ist. Durch die Textbasiertheit der IuK-Technik können keine weiteren z.B. non-verbale Signale transportiert werden, die gleiche Interessen und eine gemeinsame Kultur über eine Gruppenbezogenheit zur sozialen Bewegung übermitteln und Vertrauen generieren. Es wird besonders in der sozialen Bewegung inseriert, weswegen ein mindestens minimales Vertrauen gegenüber den potenziellen Mitarbeitern aus dieser Bewegung angenommen werden kann.

Weiterhin besteht wegen der Anonymität des Internets und mangelnder Glaubwürdigkeit eine Opportunismusgefahr, da das potenzielle Mitglied zwar sein Humankapital, seine Selbstmotivation und Zustimmung zur Lizenz (GPL) signalisieren kann, aber weder seine

Identität noch seine Absichten feststellbar sind. Auch Systemvertrauen hinsichtlich digitaler Signaturen unterstützt das Signalling nicht. Adverse Selection-Probleme sowie Information Overflow werden kaum gemildert.

Eine Milderung dieser Probleme durch Verhandlung mit Konsens ist mit hohen Instrumentnutzungskosten verbunden. Diese sind hoch, weil zuerst eine Interessensgleichheit zwischen Projekt und potenziellem Mitarbeiter über Verhandlungen hergestellt werden muss. Projektleiter und potenzieller Mitarbeiter verhandeln über die Qualität des Quellcodes, aber auch über den Zeitpunkt der Aufnahme. Auch Fragen und Probleme der Programmierung werden zwischen diesen, aber auch mit realen KDE-Mitgliedern diskutiert und verursachen hohe Kommunikationskosten. Gleichzeitig wird prozessbasiertes Vertrauen zwischen Projekt und potenziellem Mitarbeiter aufgebaut.

#### *6.1.2.4. Arbeitsplanung: Zieldefinition sowie detaillierte Entwicklungsplanung*

Beim Eintritt wird bei den Eintretenden durch den Eintrittsprozess eine gemeinsame Basis im Sinne einer Gruppen-/Projektkultur geschaffen, wobei die gruppenorientierte, u.a. auch reziproke Kooperation besonders unterstützt wird. Dieser „Restricted Access to Exchanges“ unterstützt einen Basiskonsens bzw. Grundvertrauen, die sich generell auf alle Koordinationsformen in nachgelagerten Teilprozessen auswirken. Eine Unterschiedlichkeit der Erwartungen, des Humankapitals und der Ziele wird minimiert und somit das Auftreten von initialen Transaktionsproblemen wie Information Overflows oder asymmetrische Information durch hohe Aufgabenkomplexität abgeschwächt. Dies wird beim Open Source-Projekt durch die Selbstselektion von Akteuren mit intrinsischer Motivation und weitgehend gemeinsamen Zielen aus der sozialen Open Source-Bewegung stark unterstützt.

- Bei der *Kooperation* sind die Transaktionskosten durch das bestehende Vertrauen sehr niedrig und somit diese Koordinationsform effizient.<sup>168</sup> Die Mitglieder teilen einen überwiegenden Anteil der gemeinsamen Ziele, weswegen kontroverse Diskussionen nicht zu erwarten sind und verhältnismäßig geringe Kommunikations-/Verhandlungskosten bei der Abstimmung des weiteren Vorgehens entstehen.

Gruppenbasierte Kooperation unterstützt das Bereitstellen von neuen oder passenden Ideen aus anderen Open Source-Projekten, da intrinsisch Motivierte, darunter potenzielle Mitarbeiter oder andere Projektexterne, gute Ideen unter Umgehung eines Informationsparadoxons von sich aus weitergeben, ohne dass Kosten für das Projekt entstehen (vgl. Taubert 2006). Die Kosten für die Übernahme bisher bereitstehender, bekannter Ideen aus anderen Open Source-Projekten umfasst u.a. geringe Suchkosten nach einer passenden Lösung, da mit der hohen Anzahl von projektinternen und –externen Beteiligten wahrscheinlich einer der Beteiligten davon Kenntnis hat.

Adverse Selection, aber auch Free Rider-Probleme werden gemildert, da über Verhandeln mit Konsens alle Möglichkeiten durchdiskutiert werden. Vertrauen in die ehrliche Absicht des Vorschlagenden unterstützt dies. Probleme, die andere bisher nicht sahen, oder irrelevante Informationen werden durch einen der vielen Beteiligten aufgedeckt („Vielaugenprinzip“), so dass kaum spätere Umorientierungen nötig sind.<sup>169</sup>

---

<sup>168</sup> Durch den Eintritt wird auch bei Virtualität eine gemeinsame Basis geschaffen, die sich über gruppenbasiertes Vertrauen bei einer gemeinsamen Gruppen-/Projektkultur und persönlicher Bekanntschaft der Akteure mit prozessbasiertem Vertrauen etabliert und die reziproke und gruppenbasierte Kooperationsart unterstützt (vgl. Jones, Hesterly, Borgatti 1997: 926). Das Problem der Anonymität im Internet wird zudem gemindert.

<sup>169</sup> Dies basiert auf der Feststellung, dass kooperative Zusammenarbeit in einer Gruppe bessere Entscheidungsergebnisse (höhere Lerngewinne, bessere Integration von Fachwissen, Generierung von mehr

Zusätzlich werden Information Overflows abgeschwächt, da erstens nur wenige wichtige, gesamtprojektbezogene Themen mit hoher Komplexität verhandelt werden. Ist die Komplexität der Aufgaben/Software zu hoch, so dass manche Sachverhalte nur mit hohen Kosten über die textbasierten virtuellen Kanäle kommuniziert werden können, werden in diesen seltenen Fällen die Verhandlungen in transaktionskostengünstigeren Face-to-Face-Treffen wie die alljährlichen Konferenzen fortgesetzt. Zweitens hat nicht jedes Mitglied etwas zu den Themen zu sagen, obwohl Äußerungen und Kritik im Prinzip möglich sind, weswegen keine Äußerung auf der Projekt-Mailingliste als Zustimmung gewertet wird.<sup>170</sup> Drittens werden durch die Modularisierung der IuK-Technik in Form von verschiedenen technischen Diskussionsrunden, z.B. verschiedenen Mailinglisten oder dem Chatsystem mit weniger Diskussionsteilnehmern, Information Overflows vermieden. In diesen kleinen Diskurszirkeln können Entscheidungen vordiskutiert und vorstrukturiert werden, was dann später im Plenum besprochen wird. Insgesamt werden durch eine Verringerung der Kommunikationsbeiträge zudem weitere Kommunikationskosten vermieden.

Mögliche Hold Ups, bei denen ein Mitarbeiter mit hohem projektspezifischem Humankapital für ihn nutzenbringendes durchsetzt, indem er nicht-normenkonformes Verhalten an den Tag legt oder eine Aufkündigung der Zusammenarbeit androht, können über gegenseitiges Vertrauen gemildert, aber nicht verhindert werden.

- Die mittlere Bedeutung der *Hierarchie/Macht* ist auf die geringe Senkung der initialen Transaktionskosten und der geringen Instrumentnutzungskosten zurückzuführen.

Eine Bereitstellung von neuen und passenden Ideen wird durch einen zentralen Akteur als Überlegenen kaum verbessert, eher entstehen dadurch vermehrt Transaktionskosten: Es treten zwar geringe Entscheidungskosten auf, da nur der zentrale Akteur entscheidet. Doch bestehen hohe Kommunikationskosten, da eine Entscheidung nicht unbedingt für die optimalste Lösung erfolgt, ständig Probleme gemeldet werden und nachgebessert werden müssten.<sup>171</sup> Außerdem entstehen hohe Kosten für die Suche nach passenden Ideen anderer Open Source-Projekte, da dabei materielle Anreize gesetzt bzw. Strafen angedroht und die Suche der Untergebenen ständig überwacht werden muss. Außerdem wird eine Entscheidung nicht von den intrinsisch motivierten Mitarbeitern akzeptiert, wenn deren eigene, freiwillige Beteiligungsbereitschaft gestört wird (vgl. Jung 2005: 450).

Adverse Selection durch unwichtige und falsche Informationen, aber auch Free Rider-Probleme oder Hold Ups durch wichtige, überlegene Projektbeteiligte können trotz des überwiegend gemeinsamen Interesses auftreten, da die Interessen nicht unbedingt deckungsgleich sind. Sie können jedoch durch Mitarbeiter mit hohem projektspezifischem Humankapital als zentrale Akteure aufgedeckt und somit vermieden werden. Geringe Instrumentnutzungskosten entstehen dabei ebenso wie beim Ignorieren von offensichtlich unwichtige Informationen oder Free Rider-Problemen. Kommunikative, textbasierte Macht als kollektive Sanktion entwickelt geringe Kommunikationskosten, um störende und nicht themenfokussierte Beiträge in der Mailingliste abzustrafen, d.h. manche Mitarbeiter

---

Lösungsalternativen) ermöglicht als Wettbewerb bei der Zusammenarbeit in Form eines gegeneinander Arbeitens (vgl. Wetzel 1998; Boos 1998; Sader 1991:105).

<sup>170</sup> Dahinter steht das Kalkül der Beteiligten, ob sie sich lieber an einer Diskussion inklusive individueller Verhandlungskosten oder lieber an der Softwareentwicklung beteiligen. Tendenziell steht die Softwareentwicklung als Wert an sich in der Motivationshierarchie an oberster Stelle.

<sup>171</sup> Zu allgemeine Anweisungen des Überlegenen beinhaltet zuviel Freiraum für Interpretationen, so dass dem Willen des Mächtigen nicht entsprochen wird und eine transaktionskostenerhöhende präzisierende Nachsteuerung nötig macht. Eine zu genaue Anweisung, die jeden Entwicklungsschritt präzise vorgibt, erhöht die ständigen Kommunikations- und Entscheidungskosten zwischen Mächtigem und Unterlegenem (vgl. Richter, Furubotn 1996: 370f., 379f.).

schreiben dann eine ironische, verärgerte oder beleidigende Email. Strategisch eingesetztes Lob gehört als kommunikative Macht auch dazu. Nur wenig initiale Transaktionskosten werden z.B. wegen der Angst vor diesen ironischen, verärgerten oder beleidigenden Emails von anderen Mitarbeitern reduziert, da eine weitere Machtdurchführung wegen des möglichen Verlusts intrinsischer Motivation bei dem Gemaßregelten und eines Austritts aus dem Projekt nicht möglich ist.

Zusätzlich werden die Information Overflows auch durch Projektexterne durch einen begrenzten Schreibzugriff in bestimmten Mailinglisten gemildert. Mailinglistenbetreuer als Posten vergeben einen Schreibzugriff nur an bekannte, bewährte Mitarbeiter mit z.T. hohem projektspezifischen Humankapital. Dadurch wird die Anonymität im Internet und damit Free Rider-Probleme reduziert. Die Ausrichtung der Ziele durch Projektexterne oder Mitarbeiter wird auch größtenteils vermieden. Der Erhalt eines Schreibzugriffs ist zudem ein Anreiz, sich entsprechend zurückhaltend zu verhalten und keine Free Rider-Probleme oder Hold Ups durchzuführen, was auch mit geringen Instrumentnutzungskosten verbunden ist. Wird die projektbezogene Reputation (Markennamenspezifität) für einen Hold Up missbraucht, reichen die üblichen textbasierten Machtinstrumente nicht aus. Ignorieren lässt die Meinungen als allgemein akzeptiert gelten, wiederum beleidigende Emails verschlechtern das projektexterne Bild. An dieser Stelle wird üblicherweise auf kooperatives Verhandeln zurückgegriffen (s.o.) und falsche Behauptungen richtig gestellt.

- *Wettbewerb* ist ineffizient bzw. niedrig bedeutend, da wegen fehlender Gewinne von z.B. rechtlichen Besitzansprüchen o.ä. keine Ausschreibung von Ideen mit sich gegenseitig beobachtenden Akteuren möglich ist. Weiterhin fallen bei einer Ausschreibung hohe Generierungskosten wegen redundanter Arbeiten bei den vielen Beteiligten an, die nur bedingt durch die erhöhte Qualität der Vorschläge ausgeglichen werden können. Ein Vergleich mit einer Entscheidung für eine Idee erfolgt durch eine Person oder durch eine Gruppe, wobei wiederum Kosten anfallen. Dadurch sinken die initialen Transaktionskosten nicht. Vielmehr kann eine wettbewerbliche Ausschreibung oder eine Wahl nach dem Mehrheitsprinzip an dieser Stelle zu Konflikten führen, die am Ende bei einer Eskalation die Zusammenarbeit sterben lässt.

Neben einer Ausschreibung ist eine Mehrheitswahl eine andere wettbewerbliche Art, Ideen zu generieren und auszuwählen. Bei einer Mehrheitswahl werben, anders als bei einer Ausschreibung, verschiedene vorschlagende, bestehende „Parteien“ für ihre Idee. Damit wird eine Informationsasymmetrie überbrückt, was einerseits gegen die Adverse Selection-Problematik, Free Rider-Probleme oder Hold Ups wirkt. Andererseits werden durch die Überzeugungskampagnen nicht nur die vorschlagenden Mitarbeiter von der Produktion abgehalten. Zusätzlich bewirkt eine Wahl nach dem Mehrheitsprinzip hohe Kommunikations-, Entscheidungs- und Auswahlkosten bei allen Mitgliedern, wobei aber besonders bei den vergleichsweise wenigen Vergleichsausführenden hohe Vergleichskosten anfallen.<sup>172</sup> Die Kosten werden zwar durch die luK-Technik durch eine beschleunigte und automatisierte Mehrheitswahl etwas gedämpft, doch sind sie so hoch, dass bei den vergleichbaren Open Source-Projekten Gnome bzw. Debian diese nur manchmal bis selten und nur bei sehr wichtigen Richtungsentscheidungen und zentralen Posten angewendet werden.<sup>173</sup> Dies liegt auch an dem aufwändigen Authentifizierungsprozess, der nur durch hohe Kommunikationskosten über aufwändige elektronische Kommunikation mit Fürsprechern oder wegen der weltweiten Verbreitung über aufwändige persönliche Treffen vorankommt. Au-

---

<sup>172</sup> Effizient ist der Wettbewerb, wenn wenige abzustimmende Ziele mit wenigen Zielmerkmalen und wenigen Abstimmenden vorhanden sind. Die Transaktionskostenerhöhung fällt bei einer begrenzten Anzahl von Mitarbeitern an, die die Durchführung und Auswertung der Wahl organisieren.

<sup>173</sup> s. z.B. „<http://www.debian.org/vote/>“ und „<http://foundation.gnome.org/referenda/2004-10/>“

ßerdem wird über die Mehrheitswahl ein Ziel ausgewählt und dessen Unterstützer als Gewinner bestimmt. Eine Minderheit, die ihre Interessen nicht gewahrt sieht und das Open Source-Projekt früher oder später verlässt oder schädigt, geht leer aus. Ihr Ersatz, aber auch die Sanktionen verursachen hohe Transaktionskosten.<sup>174</sup>

Ein minimaler Wettbewerb unterstützt die hierarchische Anreizsetzung durch die Schreibzugriffvergabe für Mailinglisten, indem sich die Mitarbeiter untereinander vergleichen, ihren Arbeitsaufwand steigern und Adverse Selection, Free Rider-Probleme oder Hold Ups weiter mildern, um einen Zugriff zu bekommen.

#### *6.1.2.5. Arbeitsplanung: Arbeitsverteilung von Arbeitspaketen und detaillierte Entwicklungsplanung*

Die Arbeitsverteilung von konkreten Arbeitspaketen und die detaillierte Entwicklungsplanung gehören bei dem Open Source-Projekt KDE zusammen, da die Mitarbeiter aufgrund ihrer intrinsischen Motivation beides freiwillig übernehmen. Die detaillierte Entwicklungsplanung geht dabei mit einer Koordination der Arbeitsverteilung einher, da hier die konkrete Planung bzgl. der vorhandenen Software vollzogen wird.

- Wie auch bei dem vorherigen Teilprozess ist *Kooperation* wegen des Basiskonsenses effizient und damit hoch bedeutend. Das Bereitstellungsproblem von adäquaten Mitarbeitern für eine Aufgabe wird aber nur wenig reduziert, da zwar auf themenbezogenen Mailinglisten und über Fehlerberichtssysteme auf Aufgaben aufmerksam gemacht wird, jedoch nicht alle unerledigten Aufgaben damit angesprochen bzw. dokumentiert werden können. Um weitere Mitarbeiter zu rekrutieren, wird z.T. auf umfassenderen, gesamtprojektbezogenen Mailinglisten nach Hilfe gesucht, was aber wegen der Freiwilligkeit nicht unbedingt mit Erfolg gekrönt ist.

Das gegenseitige Vertrauen, dass sich bei der selbständigen Übernahme von Aufgaben durch Mitarbeiter (Self Selection) mit entsprechendem Humankapital entgegengebracht wird, verringert Kommunikations-, Such- und Auswahlkosten und minimiert Information Overflows. Die Mitarbeiter wissen dabei meistens selbst am besten, welche Aufgaben sie sich zutrauen können.

Geringe Kommunikationskosten entstehen durch gruppenbezogene Kooperation, d.h. das unaufgeforderte Aufmerksammachen auf vakante Aufgaben wie gemeldete Fehler über das Fehlerberichtssystem oder Emails, wenn das konkrete, gemeinsame Ziel bekannt ist.<sup>175</sup> Zudem bestehen höhere Kosten der gelegentlichen Absprache im Rahmen der reziproken Kooperation z.B. wegen gemeinsamer Softwareschnittstellen oder Selbstüberschätzung. Dabei stellen Mitarbeiter irrelevante Informationen durch Nachfragen bei anderen Mitarbeitern fest. Manche Mitarbeiter übernehmen wegen der gruppenbasierten Kooperation und aus Pflichtgefühl gegenüber der Gruppe Aufgaben, die einen hohen Arbeitsaufwand umfassen oder nicht beachtet wurden. Das Vertrauen lässt zudem Free Rider-Verhalten oder Hold Ups wegen der überwiegenden Zielübereinstimmung selten auftreten.

- *Hierarchie/Macht* ist auch hier mittel effizient bzw. bedeutend, da wie oben nur eine manipulativ-kommunikative Machtdurchführung möglich ist. Die softwarebezogenen Projektleiter mit ihrem höheren projektspezifischen Humankapital können oft wegen ihres Projektüberblicks dirigierend eingreifen und das Bereitstellungsproblem mildern, indem sie Fehlerberichte an die jeweiligen bearbeitenden Mitarbeiter weitergeben bzw. Aufgaben

---

<sup>174</sup> Im schlechtesten Fall, in dem keine weiteren Mitarbeiter mehr eintreten, reduziert sich die Anzahl der Open Source-Projektbeteiligten auf eine Person.

<sup>175</sup> Wenn aber das Ziel nicht bekannt ist, wird das Ziel mit erhöhten Kosten bei der Zieldefinition bestimmt.

neuen Mitarbeitern vorschlagen. Dabei entstehen nur geringe Instrumentnutzungskosten. Auch hier werden weiterhin bestehende irrelevante Informationen auch von Projektexternen ignoriert, was mit einer geringen Senkung der initialen Transaktionskosten einhergeht. Eine kostengünstige Anreizsetzung erfolgt durch die spätere Vergabe von Schreibzugriffen zu Mailinglisten oder der späteren Übernahme eines Postens. Die Kontrolle der Aufgabenübernahme wirkt in diesem Teilprozess nicht auf die Kosten, da die Kontrolle in der Leistungserstellung und der Arbeitszusammenführung erfolgt. Hold Ups bestehen, wenn Mitarbeiter mit einer Machtstellung z.B. andere von Aufgaben abbringen und diese selbst übernehmen oder die projektbezogene Reputation gefährden. Diese Hold Ups werden wie oben beschrieben über verärgerte und beleidigende Emails sanktioniert, was nur Kommunikationskosten erfordert.

- *Wettbewerb* ist wie in dem vorigen Teilprozess ineffizient und damit niedrig bedeutend, da es weder Einflüsse auf die initialen Transaktionskosten, noch eine Erhöhung der Instrumentnutzungskosten gibt. Eine Ausschreibung für Aufgaben oder Entwicklungsideen zur Generierung von extrinsischer Motivation erfolgt wegen fehlender Gewinne von z.B. rechtlichen Besitzansprüchen und vorhandener intrinsischer Motivation nicht. Die Übernahme von schwierigeren und komplexen Arbeitspaketen zum Erhalt von Reputation, Posten oder Schreibzugriff für Mailinglisten hat einen geringen Einfluss auf die Transaktionskosten (s.o.). Eine Mehrheitswahl von Aufgaben und detaillierten Planungsideen erfolgt wegen der fehlenden Authentifizierungsprozedur, drohenden Konflikten, der möglichen parallelen Softwareentwicklung und ständiger Anpassungen der Aufgaben/detaillierten Planungsideen nicht (s.o.). Der gleichzeitige Vorschlag der Übernahme einer Aufgabe und die gleichzeitige detaillierte Entwicklungsplanung für eine Aufgabe würde zusätzlich die Redundanzen und damit die Erstellungskosten bei den Mitarbeitern erhöhen, die bei einer hohen Anzahl stark ansteigen. Außerdem entstünden dadurch nachfolgende Auswahl- und Vergleichskosten.

#### *6.1.2.6. Arbeitsverteilung von verantwortlichen Posten mit Aufgabenbereichen*

- Die Arbeitsverteilung von Posten unterscheidet sich von der Arbeitsverteilung von konkreten Aufgaben insofern, als Posten als relativ knappes Gut für das Projekt steuernd eingreifen und mit hoher Verantwortung einhergehen, aber auch einen größeren Aufgabenbereich umfassen. Im Gegensatz zur Arbeitsverteilung von Aufgaben und detaillierten Entwicklungsplanung ist bei der Postenvergabe *Hierarchie/Macht* am effizientesten bzw. bedeutendsten. Das Bereitstellungsproblem, dass Mitarbeiter wegen eines hohen Arbeitsaufwands diese Posten nicht gern übernehmen (keine Self Selection), wird darüber gelöst, dass die Übernahme eines Postens eine Ehre, d.h. eine Belohnung ist. Der bisherige Posteninhaber als Mächtiger mildert durch seine Suche nach einem Nachfolger mit hoher Reputation und hohem projektspezifischen Humankapital ein Informationsparadoxon bzw. eine asymmetrische Informationsverteilung sowie Free Rider-Probleme. Dabei kann der Posteninhaber die Mailinglisten, Homepage oder das Versionsmanagementsystem nach geeigneten Kandidaten durchforsten. Oftmals fallen bei seiner „Suche“ kaum Such-, Kontroll- und Entscheidungskosten an, da der bisherige Posteninhaber eine hohe Reputation und durch seine bisherige Tätigkeit Kenntnisse des Projekts und der Mitarbeiter besitzt. Er kann die meisten irrelevanten Informationen, falsche Identitäten o.ä. aussortieren. Außerdem werden die Aktivitäten nur von ihm ausgeführt. Hold Ups gegenüber dem suchenden bisherigen Posteninhaber bestehen, da der Posteninhaber Arbeitszeit in seine Postentätigkeit investiert hat und er ein Interesse an der Fortführung der Arbeit des Postens hat. Potenzielle Nachfolger können daher Bedingungen stellen und u.U. nach Erfüllung der Bedingung trotzdem den Posten nicht übernehmen. Diese Hold Ups treten aber kaum auf, da der bisherige Posteninhaber Macht und z.T. Zugang zu wichtiger IuK-technischer Infra-

struktur hat. Außerdem kennt er die Projektbeteiligten gut und kann die Zuverlässigkeit des möglichen Nachfolgers abschätzen oder einen anderen Nachfolger aussuchen. Weiterhin entstehen kaum Kosten durch das textbasierte Anpassen der IuK-Infrastruktur an den neuen Inhaber.

- *Kooperation* ist nur mittel effizient bzw. bedeutend, da die initialen Transaktionskosten nur wenig gesenkt und wenige Instrumentnutzungskosten entstehen. Die Bereitstellung eines möglichen Kandidaten wird durch vertrauliche Hilfe und Diskussionen mit anderen Mitarbeitern unterstützt, da der bisherige Posteninhaber keine vollständige Information über das Projekt hat. Dadurch können auch initiale transaktionskostenerhöhende Fehlentscheidungen des Posteninhabers durch Information Overflows, irrelevante Informationen oder falsche Identitäten o.ä. vermieden werden. Kandidaten, die zu Free Rider-Problemen oder Hold Ups neigen, werden durch die zusätzliche Information eher identifiziert. Die anderen Mitarbeiter mit hohem projektspezifischen Humankapital helfen, da ein gemeinsames Interesse und Vertrauen durch die Zugehörigkeit zum Projekt vorhanden ist. Zusätzlich werden die Suchkosten des Posteninhabers durch den Einbezug anderer Mitarbeiter verringert, wobei geringe Kommunikationskosten durch reziproke Kooperation zwischen Posteninhaber und einigen wenigen Mitarbeitern entstehen.

- Ein *Wettbewerb* um den Posten findet nicht statt und ist deswegen ineffizient bzw. unbedeutend, weil eine Ausschreibung mit gegenseitigem Vergleich der Bewerber oder der Aktivitäten der potenziellen Nachfolger die initialen Transaktionskosten kaum senkt. Die Bewerber würden vor allem ihre Gewinnchancen verbessern und alle weiteren Aktivitäten einstellen, was das Vorankommen im Projekt behindert. Es würden Redundanzen in den Aktivitäten bestehen, wobei sich das gleiche Bild wie bei der Arbeitsverteilung von Arbeitspaketen und detaillierten Entwicklungsplanungen ergeben würde. Damit diese Kosten nicht auftreten, wird als Kriterium die Reputation zur Auswahl verwendet, wobei die Reputationserhöhung und damit der Wettbewerb zwischen den Beteiligten in der Leistungserstellung stattfindet. Außerdem entstehen geringe Instrumentnutzungskosten durch den Vergleich der Reputation der potenziellen Nachfolger durch den bisherigen Posteninhaber oder seine helfenden Mitarbeiter.

#### *6.1.2.7. Leistungserstellung und detaillierte Entwicklungsplanung*

Die detaillierte Entwicklungsplanung erfolgt bei der Leistungserstellung, da dadurch, wie oben beschrieben, initiale Transaktionsprobleme gesenkt werden können. Zugleich bestehen ähnliche Ausprägungen und Bedeutungen der Koordinationsformen in der Leistungserstellung und der Arbeitsverteilung von Arbeitspaketen, die aber nicht unbedingt auf die detaillierte Entwicklungsplanung zurückgeführt werden können.

- *Kooperation* ist auch bei der Leistungserstellung besonders effizient bzw. bedeutend, da der Basiskonsens verbunden mit einer intrinsischen Motivation gegenseitiges Vertrauen ermöglicht. Das gemeinsame Interesse der Softwareentwicklung verhindert einen überwiegenden Teil von Shirking, Free Rider-Verhalten in Form des Einbaus von Computerviren, Spionagesoftware, beabsichtigten Fehlern in den Quellcode oder Hold Ups durch projektspezifisches Humankapital bzw. Posten. Die intrinsische Motivation im Zusammenhang mit der gemeinsamen Zielorientierung führt zu einer Art Selbstkontrolle und geringen Instrumentnutzungskosten. Zusätzlich verhindert Kooperation in der Arbeitsplanung durch ein kalkulatives Vertrauen und einen ausgehandelten Konsens Fehlentscheidungen oder Missverständnisse bei der Arbeit an Softwareschnittstellen und dadurch einen Großteil der Nachverhandlungen in Form von detaillierter Entwicklungsplanung. Eine hin und wieder auftretende gruppenbasierte Kooperation mit geringen Instrumentnutzungskosten tritt auf, wenn unaufgefordert Erkenntnisse oder Lösungsmöglichkeiten gemeldet werden. Zusätz-

lich tritt eine reziproke Kooperation mit geringen Instrumentnutzungskosten durch Nachverhandlungen oder beim Testen von Software hin und wieder auf, wenn Softwareentwicklungsprobleme bestehen.

Auch die Mitarbeiter kümmern sich weitestgehend selbst um die detaillierte Entwicklungsplanung, was zu weiteren Transaktionskostensparnissen führt. Zudem senken kleine softwarebezogene Diskurszirkel in Mailinglisten mit wenigen Mitgliedern diese Kommunikationskosten.

- *Hierarchische* Elemente werden weiterhin benötigt, da die Wahrscheinlichkeit von Nachverhandlungen durch unaufgedeckte oder Umweltveränderungen entstehende Interessengegensätze sowie Vertrauensmissbrauch (Free Rider-Problem) z.B. durch schädigenden Quellcode bestehen. Hierarchie/Macht ist aber nur mittel effizient bzw. bedeutend, da neben einer sporadischen gegenseitigen Kontrolle über das Versionsmanagementsystem eine sporadische Kontrolle der Posteninhaber steht. Diese Posteninhaber betrachten das Quellcode-Erscheinungsbild in ihrem Softwareverantwortungsbereich, da durch die ständigen Änderungen anderer ein Information Overflow entsteht. Ein Posteninhaber als Verantwortlicher löst mögliche Verantwortlichkeitsstreitigkeiten im Rahmen von Shirking, d.h. eine Aufgabe wird nicht weitergeführt, obwohl andere davon abhängig sind, oder bei lang andauernden Nachverhandlungen durch Vermittlung und schlussendlich durch ein Machtwort. Drohende kollektive Sanktionen senken zusätzlich die initialen Transaktionskosten. Dabei werden wegen der intrinsischen Motivation keine Anreizkosten benötigt. Kollektive Sanktionen werden nur bei eklatanten Störungen angewendet, z.B. wenn mutwillig Quellcode gelöscht oder geändert wird. Die Mitarbeiter können dabei nur auf kommunikativ-manipulative Macht wie Ignorieren, Drohung/Abmahnung und Ausschluss aus dem Versionsmanagementsystem zurückgreifen, wobei kaum Instrumentnutzungskosten entstehen. Also werden nur wenige initiale Transaktionskosten reduziert, wobei nur geringe Instrumentnutzungskosten bestehen. Außerdem wird dadurch die Motivation der Mitarbeiter, an der Software zu arbeiten, nicht gesenkt.

- *Wettbewerb* spielt nur als zusätzliche Motivation zur Erlangung von Reputation neben der intrinsischen Motivation eine Rolle, wobei durch eine Versuchs-und-Irrtums-Softwareentwicklung eine selbstorganisierte Humankapitalbildung erfolgt, die keine zusätzlichen Kosten verursacht. Daher bestehen mit dem Ziel der Erlangung und dem möglichen Verlust von Reputation geringe Auswirkungen auf die Entstehung initialer Transaktionsprobleme wie Shirking, Free Rider-Problem, Hold Ups und Nachverhandlungen. Bei gleichzeitiger Übernahme der Aufgaben von mehreren Mitarbeitern bestehen hohe Redundanzen, die sich in gleichzeitiger Bearbeitung eines Arbeitspakets äußern. Die Qualität der Arbeitspakete kann steigen, wird aber durch den starken Anstieg der Opportunitätskosten der Erstellung konterkariert. Die Vergleichskosten von direkten Vergleichen der Quellcodequalität oder der Anzahl der Quellcodezeilen sind hoch, da sehr viele Beteiligte in einem Open Source-Projekt zwar wenige Quellcodezeilen schreiben, aber jeweils ein standardisierter Vergleichsrahmen erstellt werden muss. Direkte Vergleiche kommen daher selten vor und erfolgen nur, wenn zwei Akteure zum gleichen Problem/Fehler fast zeitgleich Quellcode einreichen. Bei einer direkten Auswahl besteht zusätzlich die Gefahr eines Motivationsverlusts und des Austritts der wenigen Mitarbeiter mit projektspezifischem Humankapital.

#### 6.1.2.8. Arbeitszusammenführung

Die Arbeitszusammenführung unterscheidet sich von der Leistungserstellung durch das Anstreben eines Endtermins mit einem fehlerfreien Softwareprodukt als offiziellem Release. Bei der Arbeitszusammenführung ist Hierarchie/Macht effizient (s.u.), Kooperation

zwar mittel effizient, der Unterschied zwischen Hierarchie/Macht und Kooperation ist eher klein. Der kleine Abstand zwischen der Effizienz von Hierarchie/Macht und Kooperation kann darauf zurückgeführt werden, dass der Einfluss der Kooperation auf die initialen Transaktionsprobleme zwar hoch, bei manchen initialen Transaktionsproblemen jedoch geringer als in den anderen Teilprozessen ist. Zusätzlich stützen bzw. ergänzen sich beide Koordinationsformen stärker als in den anderen Teilprozessen in ihrem Einfluss auf die initialen Transaktionsprobleme. Werden initiale Transaktionsprobleme durch Kooperation nicht verhindert, wird Hierarchie/Macht eingesetzt und ggf. umgekehrt. Die leicht höhere Effizienz von Hierarchie/Macht liegt aber an dem Umstand, dass bzgl. der Termineinhaltung Kooperation zu hohe, Hierarchie/Macht dagegen niedrige Instrumentnutzungskosten beinhaltet.

- Bei *Kooperation* kann Vertrauen verbunden mit einem Basiskonsens Transaktionskosten aus Free Rider-Verhalten, Hold up und Shirking senken. Wie in den anderen Teilprozessen werden diese initialen Transaktionsprobleme stark gemildert. Free-Rider-Verhalten in Form von Computerviren, Spionagesoftware, beabsichtigten Fehlern im Quellcode etc. können bei späterer Entdeckung zu einem hohen Image-Schaden und damit hohen Kosten für das KDE-Projekt führen, da der Release stabil und fehlerfrei sein soll. Damit ist der Zustrom von potenziellen Mitarbeitern und materiellen Ressourcen gefährdet. Der Termindruck und der Anspruch, eine stabile, fehlerfreie, offizielle Softwareversion abzuliefern, macht häufige Nachverhandlungen über Problemlösungen und Fehlerkorrekturen nötig (reziproke Kooperation). Damit entstehen aber hohe Transaktionskosten, die durch einen zeitspezifischen Hold Up und Shirking weiter erhöht werden können. Der Einfluss von Shirking ist dabei etwas geringer als von Hold Ups, da für den bisherigen Bearbeiter freiwillig eingesprungen wird oder Teilaufgaben übernommen werden. Auch ein Information Overflow kann nicht verhindert werden. Auf der einen Seite tritt er wegen der Motivation der Softwareentwickler, den eigenen Softwarebereich veröffentlicht zu sehen und möglichst viel Hilfe zur Beseitigung der Fehler zu erhalten, häufig auf. Auf der anderen Seite wird ein Information Overflow durch eine gegenseitige Hilfe in den Mailinglisten der einzelnen Softwaremodule wenigstens etwas gemildert. Zusätzliche Verhandlungen über gemeinsame konsensuelle Ziele und das wiederholte Einschwören auf die Veröffentlichung als offizieller Release sind ständig nötig. Damit entstehen hohe Instrumentnutzungskosten der Kooperation.

- *Hierarchie/Macht* ist aber hoch effizient bzw. bedeutend, da die Transaktionskosten aus dem Verhandeln mit Konsens zu hoch sind. Zusätzlich kann das durch Kooperation gegebene Vertrauen missbraucht werden, aber auch Fehler können unabsichtlich z.B. beim Zusammenspiel der Softwarebereiche weiterhin bestehen. Free Rider-Verhalten wie Computerviren, Spionagesoftware, beabsichtigte Fehler im Quellcode können wie oben beschrieben übersehen werden, was zu einem Reputationsverlust bei den Projektexternen und potenziellen Mitarbeitern führen kann, von deren zukünftigem Arbeitseinsatz das Überleben des Projekts abhängt. Außerdem kann durch den Termindruck immer noch ein Hold Up entstehen, Softwareteile aufzunehmen oder abzulehnen, obwohl diese Software zentral und wichtig ist. Mitarbeiter mit hohem projektspezifischem Humankapital oder die Drohung, die projektexterne Reputation des Projekts zu reduzieren, können diesen Hold Up verstärken. Die Transaktionskosten, die dadurch entstehen würden, können nur durch eine Kontrolle und die Befürchtung einer Machtdurchsetzung vermieden werden, da ein wiederkehrender Interessensabgleich bei der Kooperation zu hohe Instrumentnutzungskosten mit sich bringt. Die drohende Letztentscheidungsmacht des Releasekoordinators über die Veröffentlichung von Softwareteilen im offiziellen Release vermeidet zu lang andauernde Diskussionen, obige Hold Ups oder Free Rider-Verhalten und fördert die Suche nach unbeabsichtigten Fehlern. Zeitspezifische Hold Ups werden vom Releasekoordinator

umgangen, indem er den Veröffentlichungstermin nach hinten verschiebt. Ein Information Overflow wird durch Ignorieren von nicht Release-relevanten Themen oder Quellcode, dem Verlust von Reputation/Ansehen, die Befürchtung und die Durchsetzung von kollektiven Sanktionen wie ironische oder ärgerliche Emails gemildert. Weiterhin wird ein Auftreten eines Information Overflows durch eine Kommunikation mit den Posteninhabern, angefangen vom Projektleiter bis hin zum Releasekoordinator, gesenkt. Eine transaktionskostenaufwändige gegenseitige Kontrolle wird vermieden, dafür kontrollieren Subprojektleiter und Releasekoordinator intensiv den Zustand ihrer Softwaremodule bzw. der gesamten Software und reagieren bei Free Rider-Verhalten wie Computerviren, Spionagesoftware, beabsichtigten Fehlern im Quellcode oder der mutwilligen, unauthorisierten Löschung von Quellcode mit Ausschluss aus dem Versionsmanagementsystem. Damit werden die initialen Transaktionskosten besonders stark gesenkt, wobei kaum Instrumentnutzungskosten auftreten.

- Ein *Wettbewerb* geringer Intensität wegen der Zeitknappheit reduziert bis zu einem gewissen Grad das Auftreten von initialen Transaktionsproblemen, indem ein Wettbewerb um geringe Fehlerhaftigkeit des Quellcodes und die Aufnahme in den Release erfolgt. Dadurch besteht eine zusätzliche Motivation neben der intrinsischen Motivation, die die Demotivation durch die häufige Kontrolle auffängt, weswegen keine weiteren Anreize gesetzt werden müssen. Ein Information Overflow wegen der großen Anzahl der hilfsbedürftigen Beteiligten verursacht dagegen einen Wettbewerb um die Aufmerksamkeit des Subprojektleiters und vor allem des Releasekoordinators, was wiederum die initialen Transaktionskosten erhöht. Eine Ausschreibung oder andere direkte Vergleiche wären wegen des Termindrucks, der vielen Fehler und zu kontrollierenden Quellcodezeilen sehr kostenintensiv. Außerdem müssten sich die Mitarbeiter entscheiden, ob sie eher Vergleiche anstellen oder lieber an Fehlerberichten arbeiten würden, anstatt an der Softwareentwicklung zu arbeiten. Zudem würde in diesem Teilprozess besonders die Auswahl eines Arbeitspakets die Kosten redundanter Arbeiten bzw. die durch den Verlust von nicht-benötigten Arbeitspaketen verursachten Kosten erhöhen, die Motivation senken und damit die nachfolgenden Kosten durch Austritte erhöhen.

### **6.1.3. Zusammenfassung: Analyse der Koordinationsformen anhand der Neuen Institutionenökonomie**

In diesem Kapitel wurden die Ausprägungen und die Bedeutung der Koordinationsformen im Open Source-Projekt KDE anhand einer Theorieinterpretation der Neuen Institutionenökonomie analysiert. Wie gezeigt wurde, kann die Theoriemodifikation, bis auf genannte Einschränkungen, für eine Analyse eines Arbeitsprozesses in einem Open Source-Projekt verwendet werden. Diese Theoriemodifikation könnte also unter Hinzuziehung der Studie von Becher et al. (2007) zu einer Theorie der Koordination in Organisationen weiterentwickelt werden. Die Analyse des Arbeitsprozesses im Open Source-Projekt KDE erfolgte in zwei Stufen, die sich jeweils in zwei kleinere Zwischenschritte unterteilen ließen:

In der ersten Stufe wurde eine Theoriemodifikation der Neuen Institutionenökonomie im Hinblick auf eine Theorie der Koordination in Organisationen durchgeführt. Bei dieser Modifikation wurde von einer Softwareentwicklung im Vordergrund und dem KDE-Projekt im Hintergrund ausgegangen. Es wurde die Grundlage der Theorieinterpretation mit den auftretenden initialen Transaktionsproblemen gelegt, die initiale Transaktionskosten verursachen. Die Koordinationsformen beeinflussen initiale Transaktionsprobleme und senken ihre initialen Transaktionskosten. Durch die Nutzung der Koordinationsformen entstehen jedoch wiederum Transaktionskosten. Die initialen Transaktionsprobleme werden vor allem durch die *Handlungsbedingungen* der Koordinationsformen, wie Vertrauen bei Kooperation, potenzieller Machteinsatz durch Anreize oder Drohung bei Hierarchie/Macht und

(gegenseitige) Beobachtung bei Wettbewerb gesenkt. Die *Handlungen* der Koordinationsformen wie Verhandlung mit Konsens bei Kooperation, Kontrolle und Machteinsatz bei Hierarchie/Macht sowie Vergleich bei Wettbewerb senken zwar auch initiale Transaktionsprobleme, generieren aber zusätzlich Instrumentnutzungskosten.

Die modifizierte Neue Institutionenökonomie konnte jedoch nicht herangezogen werden, um Vermutungen aufzustellen, warum die vorgefundenen Ausprägungen der Koordinationsformen auftreten und welche Herkunft diese haben. Dazu müssten alle möglichen auftretenden Ausprägungen mittels dieses Theoriekonzepts miteinander verglichen werden.

Stattdessen kann die Bedeutung der Koordinationsformen über einen Vergleich von rein effizienten Koordinationsformen festgestellt werden. Ausgangspunkt ist dabei, dass die vorhandenen Ausprägungen der Koordinationsformen im untersuchten Projekt die effizienten und überlebenden Institutionen sind und andere Institutionen wegen mangelnder Effizienz verschwunden sind. Durch einen Vergleich der Transaktionskosten bei den Koordinationsformen kann dann die effizienteste Koordinationsform festgestellt werden.

Die in der Einleitung zu Kapitel 6 angesprochene Problematik, dass Kooperation kaum oder schwierig in die Neue Institutionenökonomie einzubauen wäre, konnte nicht bestätigt werden. Kooperation wurde in das Theoriegefüge als Instrument zur Transaktionskostensenkung eingepasst, wobei die Unterscheidung in kostensenkende und -verursachende Anteile dies unterstützte. Mit dieser Setzung sind keine theoretischen Fragen zur Kooperation zu behandeln, die sich um eine vollständige Integration der Kooperation in dieses ökonomische Theoriekonzept mit einer möglicherweise komplizierten ökonomischen Erklärung der nicht-kalkulativen Anteile der Kooperation drehen (wie gruppenbasierte Kooperation und Altruismus).<sup>176</sup> Der grundlegende Aufbau der Theorie wurde dabei nicht verändert oder in Frage gestellt. Vertrauen kann zwar Opportunismus vermeiden und dadurch Transaktionskosten senken, aber ein Opportunismus kann trotzdem weiterhin bestehen bleiben und u.U. bei blindem Vertrauern zu einer ineffizienten Kooperation führen. Transaktionskostensenkungen werden bei Wettbewerb und Hierarchie/Macht eher durch Ausschluss von Personen aus der Transaktion, bei Kooperation hingegen durch Vertrauen auf alle bewirkt. Dadurch kann abhängig von der untersuchten Transaktion Wettbewerb und Hierarchie/Macht Transaktionskosten senken oder stark erhöhen. Das Hierarchie/Macht-Element Angst vor Strafe wurde in der Empirie wegen der Ausrichtung auf tatsächliches Verhalten nicht nachgewiesen, tritt jedoch in der Theoriemodifikation als transaktionskostensenkendes Element auf. Angst vor Strafe spielt vermutlich nur eine Rolle, wenn ein Mitarbeiter seine Ressourcen wie investiertes Humankapital oder Quellcode verlieren würde. Im KDE-Projekt gilt dies nur für wenige KDE-Mitarbeiter mit hoher Reputation und ist daher kaum anzutreffen.

In der zweiten Stufe stand das Open Source-Projekt KDE im Mittelpunkt, wobei an die vorherigen empirischen und theoretischen Erkenntnisse angeknüpft wurde. Im ersten Zwischenschritt wurden alle relevanten initialen Transaktionsprobleme in den Teilprozessen der Softwareentwicklung aufgezählt, die in einem Open Source-Projekt auftreten können. Ein Beispiel für das Free Rider-Verhalten als initiales Transaktionsproblem im KDE-Projekt

---

<sup>176</sup> Dabei ist zu vermuten, dass diese nicht-kalkulativen Anteile aus einer ökonomischen Perspektive kaum zu erklären sind und kaum in eine ökonomische Theorie integriert werden können. Um gruppenbasierte Kooperation und Altruismus in eine ökonomische Theorie zu integrieren, müsste man über den Nutzen argumentieren. Dabei wäre der Nutzenbegriff, wie in der Einleitung zu Kapitel 6 beschrieben, aufgeweicht und tautologisch geworden, da auch der Nutzen für ein soziales Aggregat oder für einen anderen Mitarbeiter mitgedacht werden müsste.

ist das Verstecken von Computerviren, Spionagesoftware oder beabsichtigter Fehler im Quellcode. Im zweiten Zwischenschritt wurden Vermutungen generiert, die die Bedeutung der Koordinationsformen begründen können. Dabei wurde über einen Vergleich der effizienten Koordinationsformen der jeweilige Einfluss einer Koordinationsform auf die Transaktionskosten analysiert. Des Weiteren wurden die Kosten der initialen Transaktionsprobleme als konstant angenommen. In einem Umkehrschluss konnte anhand der anfallenden Transaktionskosten für jede Koordinationsform die Bedeutung in einem Teilprozess der Softwareentwicklung analysiert werden. Dieser Vergleich wurde für die Teilprozesse Eintritt/Rekrutierung, Arbeitsplanung (Zieldefinition, Arbeitsverteilung von Arbeitspaketen sowie detaillierte Entwicklungsplanung), Leistungserstellung und Arbeitszusammenführung vorgenommen. Koordinationsformenelemente treten dabei an Stellen in den Teilprozessen auf, wo andere Koordinationsformen begrenzt oder ineffizient sind. Zudem zeigt die Analyse über die Neue Institutionenökonomie, dass die IuK-technische Infrastruktur im Zusammenhang mit der Koordinationsform Hierarchie/Macht Transaktionskosten senkt.

Aus der Theoriemodifikation wurden mit Hilfe der empirischen Erkenntnisse Vermutungen entwickelt, die nachfolgend vorgestellt werden. Strategische Kooperation spielt generell keine Rolle, da das informale Open Source-Projekt kein institutionenbasiertes Vertrauen erhält und aufgrund der Größe des Projekts kein kalkulatives Vertrauen entstehen kann. Außerdem lohnt sich eine Investition in kalkulatives Vertrauen nicht, da keine tauschfähigen Güter (Quellcode, Entlohnung) vorhanden sind.

Bei Eintritt/Rekrutierung ist eine Hierarchie/Machtbeziehung zwischen aktuellen und potenziellen Projektmitarbeitern am effizientesten und damit am bedeutendsten, da bei den potenziellen Mitarbeitern Kosten auftreten, um die Projektgrenze zu überwinden und aufgenommen zu werden, aber dem Projekt bzw. den Projektleitern selbst kaum Kosten entstehen. Mittel effizient bzw. bedeutend ist Wettbewerb als weitere Motivation, wobei projektspezifisches Humankapital erworben wird, um neben den aktuellen Mitarbeitern Aufmerksamkeit vom Posteninhaber zu bekommen. Kooperation hat eine geringe Effizienz bzw. Bedeutung durch die Anonymität des Internets. Es kann zusätzlich minimales Vorvertrauen zur Verfügung gestellt werden, da sonst ein Austausch zwischen den intrinsisch motivierten potenziellen und aktuellen Mitarbeitern nicht möglich wäre.

Bei Zieldefinition, Arbeitsverteilung von Arbeitspaketen mit detaillierter Entwicklungsplanung bei der Arbeitsplanung, aber auch Leistungserstellung mit detaillierter Entwicklungsplanung stellt Kooperation die effizienteste und damit bedeutendste Koordinationsform dar, da durch den Eintritt ein gemeinsames Ziel besteht und gruppenbasierte Kooperation mit hohem Vertrauen aufgebaut wurde. Verhandlungen mit Konsens werden nur bei wichtigen Problemen geführt. Hierarchie/Macht kann als eine (kollektive) Sanktion bei schädigendem Verhalten angewandt werden und ist daher mittel effizient bzw. bedeutend, wobei Kooperation ihr gegenüber dominiert. Wettbewerb ist ineffizient und hat einen geringen Stellenwert, da Ausschreibungen oder Wahlen nach dem Mehrheitsprinzip wegen redundanter Arbeiten zu kostenintensiv sind.

Die Arbeitsverteilung von verantwortlichen Posten mit Aufgabenbereichen kann durch den bisherigen Posteninhaber in Form der Hierarchie/Macht erfolgen, da dieser durch seine Mitarbeit hohe Kenntnisse über Mitarbeiter erlangt hat und daher kaum Kosten anfallen. Weitere Posteninhaber oder Mitarbeiter mit hohem projektspezifischem Humankapital unterstützen den bisherigen Posteninhaber bei der Suche, womit weitere initiale Transaktionskosten gesenkt werden können und Kooperation einen mittleren Stellenwert bei der Effizienz erhalten kann. Es werden keine Ausschreibungen durchgeführt, sondern vom bisherigen Posteninhaber die Reputation potenzieller Kandidaten verglichen, weswegen Wettbewerb ineffizient und eher von geringer Bedeutung ist.

Die Arbeitszusammenführung zeichnet sich durch eine effiziente bzw. hoch bedeutende Hierarchie/Macht aus, da über den Releasekoordinator mit dem Releaseplan und mithelfende Posteninhaber eine Fehlerfreiheit und Termineinhaltung bei einer starken Reduzierung initialer Transaktionskosten und geringen Instrumentnutzungskosten möglich ist. Durch einen gemeinsamen Basiskonsens und Vertrauen mit stark reduzierten initialen Transaktionskosten ist Kooperation mittel, aber nur etwas weniger effizient bzw. bedeutend, auch da Verhandlungen mit Konsens hohe Instrumentnutzungskosten beinhalten können und manche initialen Transaktionsproblemen nur wenig gemildert werden. Ein Wettbewerb um die Aufmerksamkeit von Posteninhabern bei Problemen führt zu hohen zusätzlichen Kosten, weswegen dieser eine geringe Rolle spielt.

## **6.2. Analyse der Koordinationsformen anhand des Neo-Institutionalismus**

Der Neo-Institutionalismus ist ein organisationssoziologisches Theoriekonzept, das sich Ende der 1970er/Anfang der 1980er Jahre erstmals formierte (vgl. z.B. Zucker 1991; DiMaggio, Powell 1991a; Meyer, Rowan 1991; Walgenbach 1999: 319). Sein Forschungsschwerpunkt liegt wie oben erwähnt vor allem auf der staatlichen, nicht-wirtschaftlichen Sphäre mit Museen, Schulen etc. oder etwas weniger häufig auf formalen, bürokratischen Organisationen der wirtschaftlichen Sphäre mit Hotels, Produktionsunternehmen etc. Der Fokus lag vor allem auf formalen Strukturen, man hat sich mittlerweile davon gelöst. Neuere Arbeiten untersuchen die „World Polity“ als nicht-formale Organisation (s. Hasse, Krücken 1999; Walgenbach 1999). Zusätzlich fixiert der mikroinstitutionalistische Ansatz als eine der Grundlagen dieses Theoriekonzepts informale Normen (s. Kap. 6.2.1.). Damit kann das eher informale Open Source-Projekt KDE mittels dieses Theoriekonzepts fokussiert werden.

Das Theoriekonzept versucht grob umrissen, die formale, z.T. informale Normenstruktur von Organisationen<sup>177</sup> zu erklären. Eine Ausrichtung auch auf Non-Profit-Organisationen ermöglicht es, wie am Anfang des Kapitels 6 schon erläutert, die Koordination der Softwareentwicklung in dem Open Source-Projekt KDE zu analysieren. Die Anpassung des Theoriekonzepts an das untersuchte Open Source-Projekt gehört dazu, gerade um die umfassenden und umfangreichen Änderungen klar und relativ detailliert darstellen zu können.

### „Alter“ soziologischer Institutionalismus

Um das theoretische Forschungsprogramm des Neo-Institutionalismus verstehen, interpretieren und modifizieren zu können, wird kurz auf seine Vorläufer eingegangen. Ein Vorläufer dieses Theoriekonzepts ist der „alte“ soziologische Institutionalismus (vgl. Hasse, Krücken 1999), der aus der deutschen Kulturanthropologie (Vertreter: Gehlen, Scheler, Plessner) und soziologischen Gesellschaftstheorien (Vertreter: Spencer, Durkheim, Parsons, Weber) besteht. In diesem „alten“ soziologischen Institutionalismus sind zwei verschiedene soziologische Ideenstränge vorzufinden, die sich entweder auf die Sanktionierung oder auf die Sozialisation als Gründe für die Einhaltung einer Norm beziehen.

---

<sup>177</sup> Im Neo-Institutionalismus wird der Begriff Organisation, entsprechend dem gängigen Verständnis, als bürokratische Organisation (wie Unternehmen und Non-Profit-Verbände) verstanden (vgl. Kap. 3.). Markt und Netzwerk als Organisationsformen sind neben der bürokratischen Organisation auch Organisationen, die hier untersucht werden und auf die der Neo-Institutionalismus auch angewendet werden kann (s.u.). In diesem Kapitel wird der Begriff Organisation konform zur Theorie und zur hiesigen Definition in Kapitel 3 verwendet.

Durkheim wie auch Tönnies zählen zu den Soziologen, die soziale Sanktionen von sozialen Gruppen als zentral für die Normeinhaltung und daher den (rechtlich) sanktionierenden Staat als Zentralinstanz und Garant für das normenbasierte Zusammenleben in den Einzelsystemen der Gesellschaft sahen (vgl. Lilienthal 2001; Durkheim 1977).<sup>178</sup> Dagegen hoben die deutsche Kulturanthropologie und Berger/Luckmann (1980) den Menschen als ein „instinktarmes Mängelwesen“ hervor, dass durch die Sozialisation ein Normensystem als zweite Natur erhält. An diesem Gedanken knüpft der Neo-Institutionalismus an.

Die Sozialisation schränkt die wählbaren Normenalternativen ein und bedingt damit die Einhaltung von (informalen) Normen. Während diese Soziologen jeweils nur einen Strang betonen, treten beide Ideenstränge zugleich in den Werken von Parsons und Weber auf. Bei Parsons (1972, 1976) definiert das internalisierte Normensystem die in einer Situation angestrebten Ziele, wobei diese Ziele wiederum nachträglich durch soziale Sanktionen z.B. vom Staat kontrolliert werden. Weber (1980: 17) sah z.B. innerliche und äußerliche Garantien (als Erwartung spezifischer äußerer Folgen), wobei die innerlichen Garantien durch Sozialisation entstehen. In Parsons und Webers Werk werden Organisationen auch als ein wechselwirkender Teil der Gesellschaft mit einer „multikontextuellen und multikausalen Einbettung“ mit informalen und formalen Normen gesehen (vgl. Senge 2005: 99f.; Parsons 1972, 1976; Weber 1980). Diese Gedanken beispielsweise von Parsons Theorie treten auch im Neo-Institutionalismus auf und können zu seiner Anpassung herangezogen werden.

Parallel zum alten soziologischen hatte sich ein alter ökonomischer Institutionalismus entwickelt (Vertreter: Veblen, Schumpeter, Polanyi, Coase, Commons etc.), der situativ und historisch Institutionen untersuchte und als Vorläufer der Neuen Institutionenökonomie (s.o.) gelten kann (vgl. Richter, Furubotn 1996: 38ff.; Göbel 2002: 48f.).<sup>179</sup> Die Neue Institutionenökonomie wurde in den Organisationswissenschaften relativ populär, woraufhin der Neo-Institutionalismus besonders als ein soziologisches Gegenprogramm zur Neuen Institutionenökonomie und eine Kritik an einer „ökonomistischen“ Ausrichtung der US-amerikanischen Organisationswissenschaft etabliert wurde (vgl. Senge 2005; Hasse, Krücken 1999: 8, 12; Fligstein 1985, 1991; Scott 2000).<sup>180</sup>

### Soziologischer Neo-Institutionalismus

Der soziologische Neo-Institutionalismus entstand aus der Erkenntnis, dass die gesellschaftliche Umwelt einer Organisation, die aus anderen Organisationen besteht, die formale Struktur dieser Organisation sowie vieler ähnlicher Organisationen beeinflusst. Die Formalstrukturen der Organisationen, die nach rationalen Gesichtspunkten ausgerichtet

---

<sup>178</sup> Die Betonung der Einhaltung der Normen durch positive und negative Sanktionen wurde auch von Dahrendorf (1961) in seiner Definition des Homo Sociologicus aufgegriffen, auf dem der Institutionalismus u.a. fußt.

<sup>179</sup> Wissenschaftshistorisch erfolgte in Ökonomie und Soziologie später eine generelle Abkehr vom Institutionenbegriff, wobei man sich in der Ökonomie auf die institutionenfreie Mikroökonomie konzentrierte und in der Soziologie eher mikro- und konfliktsoziologische Ansätze bevorzugte (vgl. Schüle 1987; Hasse, Krücken 1999: 8f.; Scott 2000). Dieser Abkehr folgte bald ein Wiederaufgreifen des Institutionenbegriffs in den beiden Disziplinen, wobei gegenüber dem alten Institutionalismus mehr oder weniger andere Schwerpunkte gesetzt wurden. In der Neuen Institutionenökonomie wurden z.B. zur Analyse von Institutionen behavioristische und mikroökonomische Elemente verwendet (vgl. Türk 2000: 125; Hasse, Krücken 1999: 9ff.).

<sup>180</sup> Die „ökonomistische“ Ausrichtung bezieht sich auf eine organisationswissenschaftliche Dominanz von Kontingenztheorie, Ressourcen-Dependenz-Ansatz, populationsökologischem Ansatz und der Neuen Institutionenökonomie, die eine Blickverengung auf Unternehmen und eine Vernachlässigung der gesellschaftlichen, nicht-wirtschaftlichen Einbindung von Organisationen mit sich brachte (vgl. Senge 2005; Hasse, Krücken 1999: 8, 12).

wurden und werden, werden beispielsweise an in der gesellschaftlichen Umwelt bestehende „Rationalisierungsmythen“, d.h. Vorstellungen von Rationalität, angepasst. Dies kann durch Sanktionierung, normativem Druck/Sozialisation oder durch Imitation erfolgen. Dabei führt „Gesellschaft“ bzw. gesellschaftliche Umwelt zu Strukturgleichheiten (Isomorphie) zwischen Organisationen (vgl. Hasse, Krücken 1999; Walgenbach 1999). Nach diesem Theorieansatz können die Koordinationsformen Kooperation, Hierarchie/Macht und Wettbewerb als rational bzw. sinnvoll erachtete Institutionen betrachtet werden, da sie sich auf verschiedene, rationale Entscheidungsmodalitäten beziehen (s. Kap. 3). Die Koordinationsformen werden aus der gesellschaftlichen Umwelt des Open Source-Projekts importiert und als Institutionen etabliert.

Der Neo-Institutionalismus griff über die US-amerikanische Rezeption von Berger/Luckmann (1980) auf die deutsche Kulturanthropologie als Teil des alten Institutionalismus zurück (vgl. Senge 2005: 109ff.; Hasse, Krücken 1999; Zucker 1991; Scott 2000).<sup>181</sup> So wurde an den Konzepten der Sozialisation und Sanktionierung zur Einhaltung von Normen und der Einbettung der Organisationen in die Gesellschaft angeknüpft. Somit liegt auch der Rückgriff auf die Gesellschaftstheorien wie von Parsons als Teil des alten Institutionalismus nahe.

Eine Kritik an dem alten soziologischen Institutionalismus ist die Übersozialisation der Normen, d.h. die Vorstellung dass den Normen freiwillig, ohne Nachdenken immer sofort gefolgt wird. Beim Neo-Institutionalismus wird weiterhin an der Sozialisation als zentralem Element des Theoriekonzepts festgehalten. Es muss den Normen allerdings nicht vollständig gefolgt werden, da verschiedene Erwartungshaltungen/Rollenerwartungen bei den Individuen bestehen und weiterhin von einer Freiheit der Wahl wie bei Parsons Systemtheorie (1972, 1976) ausgegangen wird (vgl. Hasse, Krücken 1999: 9f.; Walgenbach 1999: 338ff.). Neben der Abwendung von und den Kritiken am „alten“ Institutionalismus bestehen auch Anknüpfungspunkte zwischen „altem“ und neuem Institutionalismus. Parsons Gesellschaftstheorie kann z.B. ohne weiteres in das Theoriekonzept des Neo-Institutionalismus eingepasst werden (vgl. Senge 2005). Sie wird in dieser Arbeit verwendet, da das untersuchte KDE-Projekt als Organisation in Gesellschaftsbereiche mit unterschiedlichen gesellschaftlichen Normen bzw. Institutionen eingebettet ist, die - empirisch nachgewiesen - Einfluss auf das Projekt nehmen.

Zusätzlich kann mit einer Theorieinterpretation angesetzt werden, um die Bedeutung bestimmter Institutionen und damit auch die Kombinationen der Koordinationsformen zu analysieren. Da die Organisationen abhängig von Legitimation und externen Ressourcen sind, besteht die Möglichkeit, die Institutionalisierung bestimmter Normen aus der sozialen Umwelt über bestimmte Mechanismen (sog. Strukturangleichungsmechanismen, die zu einer strukturellen Angleichung gleicher Organisationen an gleiche Umwelten führen) bzw. über die Beschaffung wichtiger Ressourcen zu begründen (s.u.). Eine Wechselwirkung zwischen Mechanismen, Ressourcen und Umwelt lässt dabei bestimmte Institutionen in eine Organisation eingehen oder verhindert dies.

---

<sup>181</sup> Der Rückgriff auf den „alten“ Institutionalismus wurde aber im Neo-Institutionalismus kaum theoriegeschichtlich reflektiert, da sich dieses Theoriekonzept vor allem als Organisationsforschung verstand (s. DiMaggio, Powell (1991b), Scott (2000) und Senge (2005) als Ausnahmen). Der vorhergehende „alte“ Institutionalismus unterscheidet sich zwar laut Willke (1987) vom Theoriekonzept des Neo-Institutionalismus durch mangelnde theoretische Schärfe bzgl. des Institutionenbegriffs. Diese Sicht greift jedoch zu kurz, um das Entstehen des Neo-Institutionalismus zu erklären, da gerade auch beim Neo-Institutionalismus nicht von einer scharfen Definition des Institutionenbegriffs ausgegangen werden kann (vgl. Senge 2005: 20, 109ff.; Türk 2000). Eher steht bei der Abkehr von den Gesellschaftstheorien des alten Institutionalismus (wie Parsons Systemtheorie) die Abkehr von der funktionalistischen Begründung des Zusammenhalts der Gesellschaft im Mittelpunkt (vgl. Hasse, Krücken 1999).

Gerade über Sozialisation und Sanktionen als Strukturangleichungsmechanismen lassen sich die Integration und Etablierung von koordinierenden Vorstellungen in Form von Institutionen (wie Kooperation, Hierarchie/Macht und Wettbewerb) in eine untersuchte Organisation aus der gesellschaftlichen Umwelt dieser Organisation erklären. Beispielsweise können koordinierende Vorstellungen in der Gesellschaft, die in der Cultural Theory festgestellt wurden (vgl. Thompson et al. 1990), je nach sozialem Bezug in die Organisationen übernommen oder in den interorganisationalen Beziehungen nachgewiesen werden (vgl. Scott 1991: 171).<sup>182</sup>

Vorgehen bei der Interpretation der Theorie bzw. Theoriemodifizierung

Die Analyse der Ausprägung und Bedeutung der Koordinationsformen im Open Source-Projekt KDE erfolgt in zwei Stufen mit jeweils zwei Zwischenschritten.

Für eine Analyse des Open Source-Projekts KDE wird in der ersten Stufe im weiteren Kapitel 6.2.1. im Hinblick auf eine Theorie der Koordination in Organisationen aus dem Theoriekonzept eine Theorieinterpretation/-modifikation entwickelt. Mit dieser Theorieinterpretation kann das Open Source-Projekt, das bisher nicht mit dem Theoriekonzept untersucht wurde, als eher informales Untersuchungsobjekt studiert werden. Dabei können Ausprägungen und Stellenwert der Koordinationsformen im KDE-Projekt analysiert werden.

Der Autor identifiziert die einzelnen theoretischen Elemente des Theoriekonzepts, als da wären das fokussierte KDE-Projekt als Untersuchungsobjekt, die gesellschaftliche Umwelt in Form von Gesellschaftsbereichen/-subsystemen und ihren gesellschaftlichen Akteuren, Strukturangleichungsmechanismen und Ressourcen. Diese Elemente werden dabei auf das Open Source-Projekt bezogen. Im ersten Zwischenschritt werden die bisherigen Definitionen der Strukturangleichungsmechanismen und die benötigten Ressourcen untersucht und in einem zweiten die gesellschaftliche Umwelt vorgestellt, aus der die zu integrierenden Institutionen kommen. Strukturangleichungsmechanismen, Ressourcen und die Definitionen der gesellschaftlichen Umwelt werden dabei auf ihre Definitionsschwächen und -stärken hin überprüft und eigene Definitionen angestrebt, die als eigene Theorieinterpretation am Schluss dargestellt werden. Die unterschiedlichen Strukturangleichungsmechanismen und Ressourcen haben dabei für das Open Source-Projekt eine unterschiedliche Priorität, die theoretisch begründet wird und von der aus eine Bewertung der Koordinationsformen erfolgen kann. Aus dieser Gewichtung können danach die wichtigen beeinflussenden gesellschaftlichen Umwelten mit ihrem Institutionengeflecht identifiziert werden.

Die konkrete Analyse der Ausprägung und Bedeutung der Koordinationsformen im Open Source-Projekt KDE erfolgt in der zweiten Stufe in Form von Vermutungen. In einem ersten Zwischenschritt werden über wichtige Strukturangleichungsmechanismen und Ressourcen die Akteure<sup>183</sup> der Gesellschaftsbereiche mit ihren Koordinationsformenkombinationen festgestellt, die das Open Source-Projekt beeinflussen können. In einem zweiten Zwischenschritt wird, soweit dies möglich ist, die Übernahme der Koordinationsformen von den Akteuren der identifizierten Gesellschaftsbereiche in den Teilprozessen Eintritt/Rekrutierung, Arbeitsplanung (Zieldefinition, Arbeitsverteilung von Arbeitspaketen mit detaillierter Entwicklungsplanung, Arbeitsverteilung von Posten), Leistungserstellung mit detaillierter Entwicklungsplanung und Arbeitszusammenführung genauer betrachtet. Die

---

<sup>182</sup> Die Sozialisation als normenbestimmendes Element im soziologischen Neo-Institutionalismus ist dabei als Gegenpart des strategischen Handelns der Neuen Institutionenökonomie zu sehen, da eine Sanktionierung eher strategisch erfolgt (vgl. Walgenbach 1999: 320; Hasse, Krücken 1999).

<sup>183</sup> Unter Akteuren werden nicht nur individuelle, sondern auch formale und informale soziale Aggregate wie soziale Bewegungen oder Organisationen wie Open Source-Projekte, Unternehmen etc. verstanden.

IuK-Technik spielt in beiden Zwischenschritten nur bedingt eine Rolle und wird deswegen an den wichtigen Stellen erwähnt.

### **6.2.1. Theorieinterpretation des Neo-Institutionalismus - Theoretische Grundlagen der Analyse**

#### *6.2.1.1. Grundlegende Überlegungen zur Theorie*

##### Grundidee des Neo-Institutionalismus

Der Neo-Institutionalismus ist ein Theoriekonzept mit verschiedenen theoretischen Ausformungen. Ausgangspunkt des Theoriekonzepts ist die Idee, dass Organisationen sich an ihrer gesellschaftlichen Umwelt bzw. an verschiedenen Gesellschaftsbereichen ausrichten, damit sie Legitimität erhalten. Legitimität heißt in diesem Kontext, dass die Akteure an die Rechtmäßigkeit von Verhaltensweisen und gesellschaftlichen Tatsachen glauben (vgl. Weber 1980: 123). Diese Legitimität benötigen sie, um Ressourcen aus der gesellschaftlichen Umwelt zu erhalten.<sup>184</sup> Eine Legitimität wird durch Strukturgleichheit einer Organisation gegenüber anderen Organisationen erreicht, d.h. eine Organisation übernimmt die institutionellen Strukturen aus der gesellschaftlichen Umwelt über verschiedene Strukturangleichungsmechanismen. Eine Organisation „beschafft“ sich somit durch drei Mechanismen der Legitimitätserhöhung Legitimität aus der gesellschaftlichen Umwelt der Organisationen und gleicht seine Institutionenstruktur dadurch an die der anderen Organisationen aus der Umwelt an (Isomorphismus; s.a. DiMaggio, Powell 1991a).

Als Strukturangleichungsmechanismen wurden u.a. Sozialisation, Sanktion und Imitation genannt, wobei Sanktionen mit Zwang, Sozialisation mit normorientiertem Verhalten und Imitation mit einer Mimesis von vorbildhaften Organisation verbunden wird. Im Open Source-Projekt KDE haben beispielsweise die Mitglieder eine spezifische Sozialisation als Informatiker o.ä. durchlaufen. Zwang wird durch einen technischen Ausschluss ausgeübt und imitiert werden u.a. technische Vorkehrungen, die sich in anderen Projekten bewährt haben. Daneben spielen Ressourcen wie Finanzen, Personal und Anerkennung eine Rolle.

##### Mikro- und makroinstitutionalistische Ansätze

Auf diesen Kern des Neo-Institutionalismus beziehen sich makro- wie mikroinstitutionalistische Ansätze (vgl. Hasse, Krücken 1999: 18f.; Walgenbach 1999: 323ff.). Der makroinstitutionalistische Ansatz als dominierende Strömung im Neo-Institutionalismus und Ausgangspunkt dieser Theorieinterpretation betrachtet vor allem, wie der Einfluss der Umwelt auf die Organisation erfolgt, d.h. die institutionellen Regeln zwischen Organisationen. Dabei werden mikroinstitutionalistische Erkenntnisse in diese Theorie integriert. Da die Organisationen schon in bestehenden gesellschaftlichen Bezügen stehen, geht es vor allem um eine Anpassung an Veränderungen in der gesellschaftlichen Umwelt. Ein Beispiel hierfür ist die Beeinflussung der Struktur einer Organisation durch Gesetzesänderungen, Personalwechsel, Unternehmensberatungen oder Professionen (vgl. Hasse, Krücken 1999; Walgenbach 1999).

Der mikroinstitutionalistische Ansatz konzentriert sich auf die Sozialisation und wird hier dargestellt, da diese ein zentrales Element im Neo-Institutionalismus und in dieser Theo-

---

<sup>184</sup> Damit zeigt sich eine zusätzliche Übereinstimmung zwischen Neo-Institutionalismus und Parsons Gesellschaftstheorie. Beide Theorien gehen von einem Funktionalismus aus, d.h. dass alle Gesellschaftsteilnehmer nicht auf Konflikte, sondern auf die Bewahrung der funktionierenden Gesellschaft ausgerichtet sind.

rieinterpretation ist. Der Ansatz kann aufzeigen, wie sozialisierte, kognitive Skripte allein Legitimität herstellen. Zentral für den mikroinstitutionalistischen Ansatz ist ein abgeändertes sozialpsychologisches Experiment<sup>185</sup> zur Entstehung und Persistenz von Normen. Dabei stellte sich heraus, dass bei Vergleichsgruppen, die den Versuch als organisationale Situation interpretierten, die Norm aufrechterhalten wird. Sanktionen oder Anreize spielten dabei keine Rolle. Eher kann davon ausgegangen werden, dass organisationale Verhaltensweisen als kognitive Skripte vorher von den Probanden sozialisiert und durch die Interpretation der Versuchssituation als organisationale Situation abgerufen wurden (vgl. Zucker 1991).

Entkoppelung zwischen Aktivitätsstruktur und Formalstruktur: unterschiedliche Institutionalisierung durch Strukturangleichungsmechanismen

Im makroinstitutionalistischen Ansatz wurden starke Isomorphien bzgl. der Formalstruktur zwischen Organisationen aus einem Bereich festgestellt. Im Allgemeinen wurde davon ausgegangen, dass Organisationen dabei ihre Formalstruktur an die Umwelt anpassen und zugleich die damit verbundenen Verhaltensweisen übernehmen. Später fand man in empirischen Studien heraus, dass sich trotz dieser formalen Strukturgleichheiten oder der nach außengerichteten Kommunikation (talk) die Aktivitäten, u.a. informale Normen, in den Organisationen z.T. erheblich von der formalen Struktur bzw. talk unterschieden. Damit besteht nur eine „lose“ Kopplung zwischen talk, formaler und informaler (Tätigkeits-)Struktur, wobei sowohl eine starke Koppelung, als auch eine völlige Entkopplung auftreten kann<sup>186</sup> (vgl. Powell 1991, 184ff.; Hasse, Krücken 1999: 23ff., 57ff.; Meyer, Rowan 1991; Brunsson 1989; Fligstein 1985).

Die lose Kopplung bzw. Entkopplung zwischen talk und Aktivitäten bzw. formaler und informaler Struktur kommt zustande, wenn den individuellen Akteuren im Prinzip viele Erwartungen und Normen gegenüberstehen, d.h. eine Anpassung an viele unterschiedliche, auch entgegengesetzte und z.T. widersprechende Institutionen von verschiedenen Organisationen aus der gesellschaftlichen Umwelt stattfinden soll. Dadurch entsteht eine prinzipielle Freiheit der Wahl:

„Die Autonomie der Akteure, nach eigenen Maßstäben entscheiden zu können, erwächst gerade aus der Vielzahl konkurrierender Institutionen, indem diese Institutionen multiple Logiken des Handelns zur Verfügung stellen“ (Edeling 1999: 13; vgl. Powell 1991, 195).

Die prinzipielle Wahlfreiheit bedeutet, dass zwischen den Normen und Erwartungen aus der Umwelt ausgewählt werden kann, welcher gefolgt wird. Welche „Entscheidungsregel“ auf einer Metaebene legt also fest, welche Bedeutung eine Institution bekommt, so dass sie in den Aktivitäten eine gewichtige Rolle spielt. Dahinter steht die generelle Frage, welche Institutionen aus der gesellschaftlichen Umwelt in der Organisation eine zentrale Position bzw. einen hohen Institutionalisierungsgrad gegenüber den anderen Institutionen erhalten und wie sie sich als formale oder informale (Tätigkeits-)Struktur niederschlagen. Dabei muss diese prinzipielle Wahlfreiheit als eine Entscheidung für oder gegen das Übernehmen von Normen nicht durch eine strategische Entscheidung, sondern kann auch

---

<sup>185</sup> In dem sozialen Experiment sollte eine Gruppe von Testpersonen die Schwankungsbreite der optischen Täuschung des autokinetischen Effekts benennen, bei dem ein Lichtpunkt im dunklen Raum zu oszillieren scheint. In mehreren Vergleichsgruppen wurde die Einordnung der Versuchssituation verändert, so dass Versuche als keine, eher persönliche oder eher organisationale Beziehungen gewertet wurden. In eher organisationalen Beziehungen wurde eine bestimmte Schwankungsbreite über mehrere Generationen von Teilnehmern beibehalten.

<sup>186</sup> Dies wurde so interpretiert, dass die Organisation über ihre Formalstruktur bzw. nach außen gerichteten Präsentation (talk) Vertrauen verbreitet.

durch Routinen oder sozialisierte, kognitive Skripte (s.u.) bestimmt werden.<sup>187</sup> Die Entkopplung in diesen Organisationen zeigt also auf, dass bestimmte Institutionen durch den Einfluss von Strukturangleichungsmechanismen, gesellschaftlicher Umwelt und Ressourcen stärker institutionalisiert wurden als andere. Also bestimmen verschieden stark institutionalisierende Strukturangleichungsmechanismen und unterschiedlich wichtige Ressourcen mit den verschiedenen beeinflussenden Gesellschaftsbereichen, welche Koordinationsformen als Institutionen im KDE-Projekt eine hohe oder niedrige Bedeutung haben. Die empirische Momentaufnahme der Koordinationsformen des KDE-Projekts kann so erklärt werden. Aus diesem Blickwinkel ist davon abzusehen, Institutionalisierungsstärke in Abhängigkeit von Formalisierung zu betrachten. Außerdem ist das Open Source-Projekt KDE überwiegend durch informale Normen bestimmt, die unterschiedlich stark institutionalisiert wurden.<sup>188</sup>

Das Theoriekonzept besteht also aus den Grundelementen Strukturangleichungsmechanismen, Ressourcen und gesellschaftliche Umwelt, die für ein besseres Verständnis voneinander analytisch getrennt betrachtet werden. Hierdurch kann beim bisher nicht mit dieser Theorie untersuchten Open Source-Projekt KDE herausgefunden werden, welche sozialen Aggregate auf diese einwirken. Die Wichtigkeit einer Institution für das KDE-Projekt richtet sich also nach der Legitimitätsbeschaffung über unterschiedlich starke Strukturangleichungsmechanismen und der Versorgung mit unterschiedlich wichtigen Ressourcen, die in jeweils einem Unterkapitel (in Kap. 6.2.1.2. und 6.2.1.3.) zunächst einzeln betrachtet werden. Nach der Betrachtung der gesellschaftlichen Umwelt (in Kap. 6.2.1.4.) wird in Kapitel 6.2.1.5. das überarbeitete Theoriekonzept vorgestellt. Die Strukturangleichungsmechanismen sind Sozialisation, Sanktion und Imitation. Die organisationalen Ressourcen bestehen z.B. aus (finanziellen) Ressourcen, Personal und Anerkennung (vgl. Powell 1991: 195), die anhand des untersuchten KDE-Projekts überprüft werden.

#### *6.2.1.2. Strukturangleichungsmechanismen der Legitimitätserhöhung*

Zentral im Theoriekonzept des Neo-Institutionalismus sind drei Mechanismen, die für die untersuchte Organisation Open Source-Projekt bzw. für ein Cluster an Organisationen die Legitimität erhöhen und die Institutionenstruktur dieser Organisation(en) an ihre gesellschaftliche Umwelt angleichen. Dazu werden die bisherigen umfangreichen, aber z.T. unpassenden Konzepte der drei Mechanismen beleuchtet und aus ihrer Konzeption eine eigene Konzeption zur Analyse des KDE-Projekts entworfen.

#### *Bisherige Definitionen der Strukturangleichungsmechanismen der Legitimitätserhöhung*

Eine ältere und eine neuere Definition der drei Mechanismen der Legitimitätserhöhung, den sog. Strukturangleichungs- bzw. Isomorphismusmechanismen, bestehen in der Literatur (vgl. DiMaggio, Powell 1991a). Die neuere Definition greift zwar Aspekte der ersten Definition aus der Anfangszeit auf, ergänzt sie aber um die später eher verbreitete Auffas-

---

<sup>187</sup> Homo Oeconomicus als „Metaentscheidungsregel“ wurde schon in Kap. 6.1. untersucht. In diesem Kapitel spielt die „Metaentscheidungsregel“ Homo Sociologicus (s.o.) eine Rolle.

<sup>188</sup> Gerade beim Open Source-Projekt KDE sollte die Unterteilung in formale und informale Institutionen aufgegeben werden, da rechtlich relevante formale Regeln im KDE-Projekt kaum eine Rolle spielen. Außerdem erfolgt die Kommunikation im Internet schriftlich, also kann Schriftlichkeit kein Kriterium für formale Normen sein. Als weitere formale Regeln können Anweisungen auf der Website interpretiert werden, die Website kann aber relativ schnell geändert werden. Auch werden diese Regeln in der Mailingliste, den Chats und den Foren der Website nicht unbedingt angewandt. Dabei sind die Koordinationsformen Institutionen, die mehr oder minder in die Formalstruktur integriert sind, aber vor allem als informale Normen auftreten.

sung, dass manche Institutionen durch kognitive Skripte fest in den individuellen Akteuren sozialisiert bzw. verankert sind (vgl. Scott 2000; DiMaggio, Powell 1991b: 15f.; Hasse, Krücken 1999: 52f.).<sup>189</sup> Beide Definitionen haben Schwierigkeiten bei der Konzeptionalisierung der Strukturangleichungsmechanismen (s.u.) und greifen u.a. deshalb für die Analyse des Stellenwerts der Koordinationsformen zu kurz. Daher wird aus beiden Definitionen eine passende Definition der Strukturangleichungsmechanismen entwickelt, die auch für ein Open Source-Projekt geeignet ist.

- Bei der ersten Definition wird Legitimität über drei Strukturangleichungs- bzw. Isomorphismusmechanismen erzeugt. DiMaggio und Powell (1991a) sahen dabei „Coercive, Mimetic und Normative Isomorphism“ am Werk. Coercive Isomorphism bezeichnet *Sanktionen*, die durch staatliche Vorgaben in Form von Gesetzen und durch staatliche Regulierungsinstanzen (juristisches System) oder durch informale Sanktionen von anderen Organisationen durchgesetzt werden. Mimetic Isomorphism ist die aktive *Imitation/Mimesis* von als erfolgreich propagierten, adäquat erscheinenden Problemlösungsmustern anderer Organisationen unter Bedingungen hoher Unsicherheit bei der Feststellung von Zwecken und Mitteln sowie unklaren Organisationszielen, wobei Unternehmensberater eine Rolle spielen. Normative Isomorphism bezeichnet den *normativen Druck*, der durch Professionen aufgebaut wird und sich mit der Verbreitung von Professionsvertretern in den Organisationen ausweitet.<sup>190</sup> Ein besonderer Einfluss erfolgt einerseits durch die Sozialisation gleicher Vorstellungen in einer Profession und andererseits durch die Personalselektion, die durch die professionelle Ausrichtung des Personalentscheiders beeinflusst wird (DiMaggio, Powell 1991a).

Diese kurze Zusammenfassung zeigt das Problem dieser Definition auf, dass der Anpassungsmechanismus des normativen Drucks bei DiMaggio und Powell (1991a) mit der Profession gleichgesetzt bzw. miteinander verschmolzen wird. Dabei ist die Profession ein Akteur aus der Umgebung der untersuchten Organisation, der auch über eine formale Organisation und über rechtliche Schritte Einfluss nehmen kann (vgl. Freidson 1994: 68f.). Außerdem werden Professionen auch durch die genannten Strukturangleichungsmechanismen beeinflusst: „Professions are subject to the same coercive and mimetic pressures as are organizations“ (DiMaggio, Powell 1991: 152).<sup>191</sup> Die Strukturangleichungsmechanismen finden folglich in allen gesellschaftlichen Umwelten von untersuchten Organisationen Anwendung. Damit sind sie als unabhängige Elemente der Theorie zu betrachten, die aber auf die zu untersuchende Organisation, das Open Source-Projekt KDE, zu fokussieren sind. Damit wird die Eingrenzung auf die Profession aufgehoben, so dass auch andere Gruppen wie einer sozialen Bewegung mit in die Analyse einbezogen werden können, die bei einer informalen Organisation wie dem KDE-Projekt in der Umwelt auftreten (s. Kap. 5.1.).

- Die zweite Definition geht auf Scott (2000) zurück, der den Ansatz von DiMaggio und Powell reformuliert, indem er u.a. kognitive Frames/Skripte einführt. Im Ansatz von Scott

---

<sup>189</sup> Dieser Paradigmenwechsel in der Organisationstheorie erfolgt parallel zu Entwicklungen in der Psychologie und der Anthropologie (Hasse, Krücken 1999: 52f.).

<sup>190</sup> Die Betonung des normativen Drucks ist dabei das Unterscheidungsmerkmal des Neo-Institutionalismus z.B. zum Resource-Dependence- Ansatz bzw. zur Neuen Institutionenökonomie (Walgenbach 1999: 341).

<sup>191</sup> Ein weiteres Beispiel hierfür ist das Rechtssystem, das beim Sanktionsmechanismus eine tragende Rolle spielt. Wird das Rechtssystem in den Mittelpunkt einer Analyse gestellt, beinhaltet es nicht nur Strafen und Anreize, sondern auch ein System von Symbolen, kognitiven Skripten und moralischen Prinzipien, die von Menschen, besonders von Juristen, internalisiert werden. Weiterhin werden organisationale Verfahren, die informal in Organisationen erfolgreich verfolgt werden, imitiert und als Standards in Gesetze gegossen (vgl. Edelman, Suchman 1997; Walgenbach 2002: 168f.).

(2000: 52) tragen drei Pfeiler eine Institutionen, d.h. jede Institution wird dabei durch je einen regulativen, normativen und kognitiven Mechanismus gestützt. Der *regulative* Grundpfeiler ist der Teil der Institution, der Handlungen begrenzt und reguliert, d.h. durch Regelsetzung, Überwachung und Sanktionierung die Einhaltung gewährleistet. Dies kann durch rechtliche Maßnahmen der Polizei, aber auch durch informale wie Beschämen, Ignorieren, etc. erfolgen (Scott 2000: 51f.). Damit gleicht die regulative Säule dem obigen Sanktionsmechanismus von DiMaggio und Powell (1991a). Die normative und kulturell-kognitive Säule sind aber neu definiert worden.

Die *normative* Säule beschreibt die vorschreibende, bewertende und verpflichtende Dimension des sozialen Lebens. Erwartungen anderer individueller Akteure bestimmen die Handlungen, die als Verpflichtungen empfunden werden. Dabei sind Werte als Bewertungssysteme mit angestrebten, wünschenswerten Ziele sowie Normen als konkret festgelegtes Verhalten über Zwecke und Mittel gekennzeichnet. Wie aber der obige Mechanismus Imitation mit dieser Säule zusammenhängt wird in der Literatur nicht hinreichend analysiert.

Die *kognitive* Dimension beschreibt das symbolische und kulturelle System als objektiv und außerhalb des Akteurs stehend, das akteursinterne Interpretationsprozesse einrahmt und formt. Es bildet den Rahmen für die Wahrnehmung der sozial konstruierten „Wirklichkeit“. Diese kognitive Dimension bezieht sich wie im mikroinstitutionalistischen Ansatz auf nicht-hinterfragte kognitive Skripte bzw. kulturellen „Frames“ oder Verfahrensmodelle, die wiederholte habitualisierte Routinen oder repetitive Aktionsmuster darstellen (Scott 2000: 57f.).<sup>192</sup> Diese kognitiven Skripte werden sozialisiert, d.h. werden unreflektiert übernommen, was u.a. bei der Sozialisation in eine Profession geschieht. Sie unterscheiden sich von der Mimesis als eine aktive, reflektierte Übernahme von adäquat erscheinenden Vorbildern bei DiMaggio und Powell (1991a). Diese Definition grenzt sich dadurch von den Sozialisationstheorien mit ihrem emphatischen Identifikationsbegriff bei der Normeninternalisierung ab (vgl. Hasse, Krücken 1999: 52).

Diese Reformulierung der Strukturangleichungsmechanismen von Scott krankt daran, dass diese Einteilung an Merkmale vergleichbar von Motiven gebunden ist, die im Individuum liegen. Ob einer Norm wegen ihrer Zweckmäßigkeit, Angst vor Sanktionen, Angemessenheit oder unhinterfragten Routinen gefolgt wurde, kann oftmals nicht eindeutig gesagt werden. Gerade die normative Säule lässt sich kaum fassen und kaum von den beiden, griffiger definierten, anderen Säulen trennen (vgl. Hirsch 1997: 1713). Beispielsweise treten Erwartungen in allen drei Säulen auf. In der regulativen Säule bestehen wie in der normativen Säule Erwartungen, da Sanktionen eine Reaktion auf enttäuschte Erwartungen sind. An die kognitiven Skripten sind auch Erwartungen gebunden.

Das beobachtete Verhalten im KDE-Projekt kann also nicht ohne weiteres einer Säule zugeordnet werden. Auch aus psychologischer Perspektive kann das nach außen ähnlich auftretende Verhalten nicht unterscheiden werden (vgl. Hirsch 1997: 1720). Diese Problematik führt dazu, dass die empirisch auftretenden Institutionen nicht auf ihre Ursache, d.h. der entsprechenden Säule, rückbezogen werden können.

#### Eigene Definition der Strukturangleichungsmechanismen der Legitimitätserhöhung

Im Weiteren werden die Gedanken aus beiden Definitionen von DiMaggio und Powell (1991a) sowie Scott (2000) als Bausteine für den zu erstellenden Analyserahmen übernommen, die für das Untersuchungsobjekt Open Source-Projekt geeignet sind. Als

---

<sup>192</sup> Sie fanden als paradigmatische Vorstellung parallel auch in anderen Organisationstheorien, der Psychologie und der Anthropologie Eingang (Hasse, Krücken 1999: 52f.).

ein Fazit aus der Auseinandersetzung mit den oben genannten Mechanismen folgt, dass Strukturangleichungsmechanismen und auch Ressourcen analytisch von der gesellschaftlichen Umwelt bzw. deren Akteuren zu trennen sind. Alle drei Strukturangleichungsmechanismen und Ressourcen können in den Beziehungen zwischen Organisation und jeweiliger gesellschaftlicher Umwelt auftreten.

Bei der Definition der Strukturangleichungsmechanismen wird von individuellen Akteuren als Vertreter einer Organisation wie dem Open Source-Projekt ausgegangen, die für die Organisation Legitimität beschaffen. Dabei sollte das Verhalten eines Akteurs einem Strukturangleichungsmechanismus zugeordnet werden können (s.o.; vgl. Hirsch 1997), damit überhaupt die vorgefundenen empirischen Ergebnisse diesen zugeordnet und analysiert werden können. Folglich ist von der ersten Definition von DiMaggio und Powell (1991a) mit ihrem Imitations- und Sanktionsmechanismus auszugehen, die eher auf das Open Source-Projekt angewandt werden kann. Sie wird durch die Definition von Scott (2000) ergänzt, aus der die umfassendere Idee der sozialisierten kognitiven Skripte den Professionsbezug ablösen kann.

Bei der Neudefinition wird auf Gemeinsamkeiten bei den beiden Definitionen geachtet, aber auch untersucht, ob die Mechanismen in empirischen Erkenntnissen über das Open Source-Projekt KDE nachgewiesen werden können oder Bezüge zu Koordinationsformen bestehen. Im Folgenden werden die drei Strukturangleichungsmechanismen Sanktion, Sozialisation und Imitation für eine spätere Analyse definiert und nacheinander betrachtet.

- Der *Sanktionsmechanismus* wird bei DiMaggio und Powell (1991a) sowie Scott (2000) ähnlich definiert, auch da die Definition in der Soziologie (s.o.) stark verbreitet ist. Unter dem Strukturangleichungsmechanismus *Sanktion* als regulativem Mechanismus ist die Regelsetzung, Kontrolle und Sanktionierung zu verstehen, die Erwartungen festlegen und aufrechterhalten. Dies kann einerseits durch formale, rechtliche Bestimmungen als kodifizierte Erwartungen oder über Gerichte, aber auch durch informale, nicht-kodifizierte Regeln erfolgen. Werden die Erwartungen von der organisationalen Bezugsgruppe oder den Akteuren missachtet, treten Sanktionen auf. Sowohl Sanktionen wie der Entzug technischer Zugriffsrechte als auch die kodifizierten Regeln wie Softwarelizenzen sind im KDE-Projekt nachweisbar, die Internalisierung der Normen aus Angst vor Strafe dagegen kaum. Sanktionen können also auch den Entzug beispielweise von materiellen oder finanziellen Ressourcen beinhalten, auf die ein Open Source-Projekt oder ein Mitarbeiter des KDE-Projekts angewiesen ist.

Bei Betrachtung der Koordinationsformen fällt auf, dass zwischen den Sanktionen bzw. der Sanktionierung in der gesellschaftlichen Umwelt des KDE-Projekts und der untersuchten Koordinationsform Hierarchie/Macht innerhalb des Projekts unterschieden werden muss. Der Blick der Untersuchung ist auf die organisationsinterne Softwareentwicklung gerichtet, weswegen Hierarchie/Macht als Institutionen vom reformulierten Neo-Institutionalismus betrachtet werden kann. Der Sanktionsmechanismus aus der sozialen Umwelt des Projekts gehört dagegen zum Theoriekonzept und kann einen Einfluss auf die Gestaltung der organisationsinternen Softwareentwicklung beschreiben.

- Der zweite Mechanismus ist die *Imitation*. Imitation ist die aktive, absichtsvolle Übernahme von angemessenen, vorbildhaften, erfolgreichen und legitim erscheinenden Problemlösungsmustern (Best Practice) bzw. Verfahrensmodellen, wozu auch angemessene Personalselektionsverfahren gehören. Diese Verfahrensmodelle werden aus der entsprechenden gesellschaftlichen Umwelt der Organisation übernommen, wenn unspezifizierte Erwartungen bzw. Unsicherheiten gegenüber unklaren Organisationszielen, unklaren Umwelteinflüssen und unklaren, neuen Strukturnormen bestehen. Damit wird von der Definition der normativen Säule von Scott (2000) Abstand genommen, aber deren Logik der An-

gemessenheit (Appropriateness) übernommen und auf den Imitationsmechanismus von DiMaggio und Powell (1991a) rekurriert. Vertreter des Open Source-Projekts KDE entscheiden dabei z.T. selbst, welche adäquat erscheinenden Normen oder technischen Werkzeuge wie Editor, Versionsmanagementsystem diese z.B. von anderen Projekten übernehmen. Weiterhin impliziert der Imitationsmechanismus, dass zentrale Projektmitglieder über eine Übernahme von Normen entscheiden und dies hierarchisch durchsetzen (vgl. Fligstein 1985, 1991). Dabei kann dies auch durch eine Gruppe im Projekt geschehen und in kooperativer Zusammenarbeit übernommen werden.

- Als dritter Strukturangleichungsmechanismus ist die *Sozialisation* von kulturell-kognitiven Routinen zu nennen, die sogenannten kognitiven Skripte bei Scott (2000). Die Sozialisation von Normen erfolgt nicht nur durch die Profession, wie bei DiMaggio und Powell (1991a). Die kognitiven Skripte werden bei der Sozialisation in eine jeweilige Gruppe oder ein jeweiliges soziales Aggregat verinnerlicht, wobei diese Internalisierung durch unreflektierte Nachahmung ohne Identifikation mit Gruppe/Aggregat erfolgt. Die kognitiven Skripte werden von vielen z.B. einer Gruppe, Organisation, Gesellschaftsbereichen oder Nation geteilt und sind so unabhängig vom einzelnen Akteur, bestimmen aber dessen Handlungen. Eine Zuordnung der Verhaltensweisen kann erfolgen, wenn wie in diesem Fall zentrale Personen oder ein großer Anteil der Open Source-Projektmitarbeiter vor und während der Open Source-Softwareentwicklung anderen Gruppen, Organisationen, Gesellschaftsbereichen etc. angehört haben. Beispielsweise haben viele Mitglieder eines Open Source-Projekts einen technischen Studiengang an einer Universität absolviert. Bei der Aufnahme in eine Organisation spielt zusätzlich eine unbewusste Personalselektion anhand der sozialisierten Skripte eine Rolle.<sup>193</sup> Alle drei Koordinationsformen können mittels kognitiven Skripten sozialisiert werden, weswegen dieser Mechanismus keinen theoretischen Bezug zu den Koordinationsformen hat. Dies ist mit ein Grund, warum die Sozialisation in der Theorieinterpretation im Vordergrund steht.

Weiterhin unterscheiden sich die definierten Strukturangleichungsmechanismen in der Stärke der Strukturangleichung bzw. Institutionalisierung.

#### Stärke der Institutionalisierung durch Strukturangleichungsmechanismen der Legimitäts- serhöhung

In der makroinstitutionalistischen Literatur wurde bisher davon ausgegangen, dass die drei genannten Strukturangleichungsmechanismen den gleichen Einfluss auf die Übernahme von Institutionen haben. Die Strukturangleichungsmechanismen entsprechen Transmissionsriemen, über die das Institutionengeflecht des Untersuchungsobjekts KDE-Projekt in Wechselwirkung mit seiner Umgebung tritt. Zugleich ist das Institutionengeflecht des KDE-Projekts eingebettet in die sozialen Zusammenhänge einer Gesellschaft.<sup>194</sup> Über die

---

<sup>193</sup> Die erwähnte Personalselektion hängt mit einer Sozialisation zusammen (vgl. Leuze 2007), stellt aber auch eine bewusste Entscheidung dar (Jung 2005: 147ff.). Dabei geht es um eine bewusste Entscheidung für oder gegen die organisationale Aufnahme von Personen mit bestimmtem Sozialisationshintergrund, die bestimmte (organisationelle) Kriterien erfüllen und z.B. im KDE-Projekt zu den anderen Projektbeteiligten passen sollen. Der gemeinsame Sozialisationshintergrund als Entscheidungsgrundlage wird daher unter dem Strukturangleichungsmechanismus Sozialisation subsumiert ohne die Art der bewussten Entscheidung zu betrachten. Die Art der bewussten Entscheidung, d.h. ob Personalauswahlgespräche oder Assessment center verwendet werden, wird unter den Strukturangleichungsmechanismus Imitation gezählt, da diese Entscheidungsverfahren von anderen Organisationen übernommen werden können.

<sup>194</sup> Dieses Institutionengefüge ist für einige Organisationen gleich bzw. eher ähnlich, da die institutionellen Einflüsse zwar ähnlich sein können, aber durch leicht andere Gewichtungen bei einer Organisation selten vollkommen gleich sind (Powell 1991: 195; Scott 1991: 172; Walgenbach 2002: 171ff.).

Strukturangleichungsmechanismen wird das Institutionengeflecht im Untersuchungsobjekt Open Source-Projekt ständig reproduziert bzw. immer wieder aufgefrischt (vgl. Walgenbach 1999: 322). Aus diesem Grund steht das Institutionengeflecht des Open Source-Projekts im Gleichgewicht mit der gesellschaftlichen Umwelt. Nach Scott (2000) verkeilen dabei die drei Strukturangleichungsmechanismen die Institutionen im Institutionengeflecht des Open Source-Projekts. Stehen sich aber konfligierende Normen gegenüber, kann nicht von einer Institutionalisierung ausgegangen werden, da nicht klar ist, welcher Norm zu folgen ist.<sup>195</sup> Je stärker sich die Institutionen des Projekts mit seiner Umwelt im „Einklang“ bzw. „Gleichgewicht“ befinden, desto weniger konfligierende Einflüsse auch aus den verschiedenen Gesellschaftsbereichen bestehen und desto stärker sind diese durch die Strukturangleichungsmechanismen und Ressourcen „verkeilt“ bzw. festgefügt.

Unvermeidliche Veränderungen in der gesellschaftlichen Umwelt ändern das Institutionengefüge und konfligierende Normen treten auf. Diese Veränderungen beeinflussen in dieser Idealvorstellung direkt über alle zugleich wirkenden Strukturangleichungsmechanismen das Open Source-Projekt. Dies würde zu einer ständigen Institutionalisierung und De-Institutionalisierung von Normen führen, was sich aber nicht mit der empirischen vorgefundenen, langsamen Veränderungsgeschwindigkeit von institutionalisierten Normen vereinbaren lässt. Folglich werden die gesellschaftlichen Umweltveränderungen abgefedert und das Problem der konfligierenden Normen gelöst, indem die Strukturangleichungsmechanismen unterschiedlich stark die Normen von der gesellschaftlichen Umwelt in die Organisation übermitteln. Die drei Strukturangleichungsmechanismen übermitteln Institutionen unterschiedlich stark, da Projektmitarbeiter als Individuen auf die Mechanismen unterschiedlich Einfluss nehmen können.

Die Institutionen oder Normen, die über einen Strukturangleichungsmechanismus in das Open Source-Projekt verankert werden, sind dabei je nach Mechanismus den Eingriffen der übermittelnden, individuellen Akteure ausgesetzt und dadurch unterschiedlich stabil/resistent gegen situative und kurzfristige Veränderungen. Folglich ermöglichen die Strukturangleichungsmechanismen den Institutionen eine unterschiedlich hohe Resistenz gegen eine De-Institutionalisierung (vgl. Schüle 1987: 141f.; Zucker 1991; Hasse, Krücken 1999: 19; Weber 1980:16). Die Strukturangleichungsmechanismen werden im Folgenden auf ihre Veränderungsresistenz gegenüber einem Individuum überprüft, d.h. in welchem Ausmaß diese (am wenigsten) von einem Individuum kurzfristig verändert werden können.

- Der Mechanismus *Sozialisation* bewirkt die stärkste Institutionalisierung, da die Routinen/Verfahren/Normen (kognitive Skripte) fast unhinterfragt internalisiert wurden und deswegen recht resistent gegen eine Veränderung durch ein Individuum sind. Auf die Routinen bzw. Normen ist kein direkter Zugriff möglich, da diese im Allgemeinen gar nicht erkannt werden (Scott 2000: 61; Trice, Beyer 1993: 194). Eine Situationsinterpretation bewirkt, dass diesem Mechanismus gefolgt wird, wie aus den Erkenntnissen des mikroinstitutionalistischen Ansatzes bekannt (s. Kap. 6.2.1.1.).

- Beim Mechanismus *Sanktion* werden Institutionen von außen durchgesetzt oder Erwartungen von außen an die individuellen Akteure herangetragen, wobei eine Sanktion über das juristische System oder andere Organisationen erfolgt. Dieser Mechanismus führt zu einer mittleren Stärke der Institutionalisierung, da grob umrissen ein Einfluss auf

---

<sup>195</sup> Bestes Beispiel für den Umgang mit konfligierenden Normen ist die oben genannte Entkoppelung zwischen Aktivitätsstruktur und Formalstruktur bzw. *talk*, die in empirischen Untersuchungen festgestellt wurde. Diese Entkopplung deutet darauf hin, dass spezifische Gewichtungen bei den Mechanismen vorgenommen wurden (s. Kap. 6.2.1.1.).

einer mittleren Stärke der Institutionalisierung, da grob umrissen ein Einfluss auf das juristische System nur indirekt über Überzeugungsarbeit im politischen System als Gesetzgeber erfolgen kann, wobei die kodifizierten Erwartungen im Gesetz verändert werden müssen. Zusätzlich werden Sanktionen nur im Rahmen des Rechtswegs durchgeführt, d.h. wenn der Staat oder ein Kontrahent ein Interesse daran hat. Der Einfluss eines Individuums auf Gesetze und den Gesetzgeber ist wegen eines hohen Aufwands gering, aber immerhin eher möglich als bei der Sozialisation. Der informale Sanktionierung durch andere Organisationen oder soziale Aggregate kann auch nur durch Überzeugungsarbeit begegnet werden, wobei gegenüber dem vorigen ein geringerer Aufwand zu Erfolg führen kann.

- Der *Imitationsmechanismus* geht von einem entscheidungsberechtigten Individuum oder von mehreren Mitgliedern im KDE-Projekt aus, die sich je nach Situation für die Übernahme von Institutionen entscheiden. Die Imitation institutionalisiert mit der geringsten Stärke, da eine Entscheidung unmittelbar von einem oder einigen wenigen Akteuren erfolgt und diese recht schnell wieder rückgängig gemacht werden kann. Eine Durchsetzung von neuen Normen, auch in der Aktivitätsstruktur, funktioniert an den Stellen am besten, an denen Sozialisation und Sanktion keine Institutionen unterstützen.

Zusammengefasst nimmt der Einfluss auf den Imitations- über den Sanktions- zum Sozialisationsmechanismus ab bzw. die Stärke der Institutionalisierung zu.

### *6.2.1.3. Ressourcen und ihr Einfluss auf die Institutionalisierung der Normen*

Auch Ressourcen haben einen Einfluss darauf, wie stark Institutionen aus der gesellschaftlichen Umwelt der Organisation institutionalisiert werden. Grundlage ist eine analytische Trennung der Strukturangleichungsmechanismen und Ressourcen. Bisher wurden Ressourcen kaum im Theoriekonzept des Neo-Institutionalismus tiefer diskutiert. Bei DiMaggio und Powell (1991a) stehen sie z.B. als Personal vor allem bei der Rekrutierung von Professionsmitgliedern im Hintergrund. Am ehesten treten Ressourcen als finanzielle Ressourcen oder Legitimität im Rahmen des gesellschaftlichen Sektors auf (s.u.; vgl. Scott, Meyer 1991). Eine weitere Erwähnung finden organisationale Ressourcen z.B. als (finanzielle oder materielle) Ressourcen, Personal und Anerkennung bei Powell (1991: 195), wobei Personal auch eine Ressource für eine Organisation ist (Jung 2005: 4). Diese Ressourcen dienen als Orientierung, da sie fast alle möglichen Ressourcen beim untersuchten Open Source-Projekt KDE abdecken. Bei der Softwareentwicklung im KDE-Projekt wären noch Software bzw. geschriebener Quellcode als Ressourcen zu nennen.

Ressourcen können neben den Strukturangleichungsmechanismen die Wahlfreiheit bei konfligierenden Normen einschränken, da die untersuchte Organisation wie das Open Source-Projekt von diesen Ressourcen abhängig ist (vgl. Powell 1991: 195). Das Open Source-Projekt als Beispiel richtet sich also nach der gesellschaftlichen Umwelt aus, auf deren Ressourcen sie angewiesen ist.<sup>196</sup>

Ausgangspunkt einer Bewertung der Ressourcen sind die Legitimationsgeber, die mit ihrer Ressource freiwillig das untersuchte Open Source-Projekt KDE unterstützen. Dabei können die jeweiligen Ressourcen gewichtet und in eine Rangfolge gestellt werden, die bestimmt, inwiefern sich die Legitimationsgeber beteiligen und vollständig involviert sind (im Sinne eines Commitment). Je mehr sie sich beteiligen, desto mehr identifizieren sich die Legitimationsgeber z.B. mit dem KDE-Projekt und desto glaubwürdiger ist es. Die stärkste Beteiligung erfolgt durch eine aktive Beteiligung eines Legitimationsgebers im Projekt. Die

---

<sup>196</sup> Keine zwei Organisationen benötigen die exakt gleichen, aber u.a. ähnliche Ressourcen, die sich zudem in ihrer Bedeutung für die Organisationen unterscheiden (Powell 1991: 195).

Legitimationsgeber sind damit potenzielles und tatsächliches Personal, das sich wie auf dem Arbeitsmarkt für eine legitime Organisation frei entscheidet und dort arbeitet. Dies gilt auch für Arbeitszeit oder Arbeitsergebnisse, die bei freiwilliger Arbeit auch eine Rolle spielen, da sich der Legitimationsgeber dabei mit dem Projekt auseinandersetzen muss.

Eine mittlere Bedeutung haben finanzielle bzw. materielle Ressourcen, da diese einen „Verlust“ beim Legitimationsgeber darstellen. Ein „Verlust“ an diesen Ressourcen geht zum einen mit einer Einschränkung der „Bewegungsfreiheit“ bzw. Verhaltensmöglichkeiten beim Legitimationsgeber einher, was zumindest eine geringfügige Beteiligung darstellt. Zum anderen muss er sich nicht mit dem hier untersuchten KDE-Projekt all zu sehr auseinandersetzen.

Die geringste Bedeutung hat die Ressource Anerkennung, da sich der Legitimationsgeber nicht mit dem Open Source-Projekt KDE näher beschäftigen muss. Es werden Produkte des Projekts wie Software (Quellcode) betrachtet, wobei sich die Anerkennung beim Legitimationsgeber bildet. Das Projekt erfährt von der Anerkennung erst über Dritte oder durch anerkennende Aussagen eines Legitimationsgebers. Damit hat auch die Ressource Software (Quellcode) eine geringe Bedeutung.

Ressourcen haben damit keinen Bezug zu strategischen Entscheidungen, da die Ressourcen und ihre Wichtigkeit für die untersuchte Organisation KDE-Projekt abhängig von den Legitimationsgebern aus der gesellschaftlichen Umwelt sind.

Für eine endgültige Feststellung, welche Institutionen aus der gesellschaftlichen Umwelt in das Open Source-Projekt KDE integriert werden, werden oben genannte Strukturangleichungsmechanismen und Ressourcen herangezogen. Zusätzlich kann die Bedeutung einer Institution anhand von wichtigen Ressourcen und von Strukturangleichungsmechanismen mit hoher Institutionalisierungsstärke festgestellt werden. Dies kommt einer Verkeilung nicht nur durch Strukturangleichungsmechanismen, sondern auch durch Ressourcen gleich.<sup>197</sup>

Auch wenn Ressourcen und Mechanismen analytisch getrennt betrachtet werden, bestehen doch aus theoretischer Perspektive Zusammenhänge. Die sozialisierten kognitive Skripte der Ressource Personal wirken durch den Mechanismus Sozialisation. Finanzielle und materielle Ressourcen werden hauptsächlich durch den Sanktionsmechanismus beeinflusst.

Das nachfolgende Unterkapitel beschäftigt sich mit der gesellschaftlichen Umwelt, die eine Organisation beeinflusst, das Institutionengeflecht dieser Organisation bestimmt und in der später auch die Organisation verortet werden kann.

---

<sup>197</sup> Wie die wechselnde Priorität verschiedener gesellschaftlicher Umwelten durch das Zusammenspiel von Ressourcen und Strukturangleichungsmechanismen die institutionelle Ausrichtung einer untersuchten Organisation beeinflusst, wurde im folgenden Beispiel betrachtet. Die dänische Copenhagen Business School hat sich im Lauf ihrer historischen Entwicklung bis heute an verschiedene Anforderungen für einen jeweiligen Zeitraum aus gesellschaftlichen Teilsystemen wie Wissenschaft (Universität), staatliche Verwaltung, Wirtschaft und Gemeinschaft (Studenten) angepasst. Zentral waren nicht nur finanzielle Ressourcen, die abwechselnd von der Wirtschaft sowie von Studenten und dem Staat kamen und dadurch die institutionelle Struktur der Business School beeinflussten. Auch einzelne Strukturangleichungsmechanismen z.B. gesetzliche Sanktionen durch das Bildungsministerium bestimmten den Wechsel der institutionellen Struktur. Die Autoren sprechen dabei zwar von Organizational Fields, diese decken sich aber größtenteils mit Subsystemen in Parsons Gesellschaftseinteilung (vgl. Borum und Westenholz 1995).

#### 6.2.1.4. Gesellschaftliche Umwelt einer untersuchten Organisation

In den vorangegangenen Unterkapiteln wurden die Strukturangleichungsmechanismen der Legitimitätserhöhung und die Ressourcen als erster Zwischenschritt betrachtet. An dieser Stelle wird als zweiter Zwischenschritt die gesellschaftliche Umwelt betrachtet, da aus dieser die zu integrierenden Institutionen übernommen werden. Die Herkunft der Normen bzw. Institutionen im KDE-Projekt steht also hier im Zentrum.<sup>198</sup>

Zwar sind aus der empirischen Untersuchung des KDE-Projekts einige gesellschaftliche Umwelten bekannt, doch bisher bestehen keine Erkenntnisse über die Art und die Stärke ihres Einflusses sowie die übernommenen Koordinationsformen als Institutionen. In diesem Kapitel soll daher eine Herangehensweise zur Isolierung der beeinflussenden Akteure der gesellschaftlichen Umwelt entwickelt werden, um diesen Einfluss später untersuchen zu können. Dabei spielen Strukturangleichungsmechanismen und Ressourcen eine Rolle.

Damit wird nur die gesellschaftliche Umwelt des hier untersuchten Open Source-Projekts betrachtet, weswegen im Folgenden die zwei bisherigen Konzeptionen „organisationales Feld“ und „gesellschaftlicher Sektor“ auf ihre Tauglichkeit für das Projekt überprüft werden. Da einige Umwelten des KDE-Projekts eher keine formalen Organisationen sind, sondern informale Zusammenhänge wie soziale Bewegungen oder spezifische soziale Gruppen, werden diese Konzeptionen verworfen. Die herangezogene Neo-Institutionalismus wird um Parsons Gesellschaftstheorie erweitert, um die gesellschaftliche Umwelt mit ihren Subsystemen und gesellschaftlichen Akteuren adäquat identifizieren zu können.

#### Organisationales Feld und gesellschaftlicher Sektor

Die gesellschaftliche Umwelt, aus denen die Normen übernommen werden, wurden im Neo-Institutionalismus als organisationales Feld und gesellschaftlicher Sektor konzeptionalisiert. Diese beiden Ansätze verbinden gesellschaftliche Bereiche mit Strukturangleichungsmechanismen und Ressourcen und konzentrieren sich auf einen gesellschaftlichen Nahbereich der untersuchten Organisation.

- Beim Ansatz des organisationalen Feldes bestehen Organisationen, die miteinander auf unterschiedliche Weise verbunden und ähnlichen Bedingungen ausgesetzt sind. Dabei wird ein organisationales Feld nicht nur durch direkt miteinander interagierende Organisationen, sondern auch durch die Strukturangleichungsmechanismen Imitation, normativer Druck und Sanktion bestimmt (DiMaggio, Powell 1991a: 155f.). Die Ressourcen kommen nicht nur von Organisationen mit geografischer Nähe, sondern vor allem von Organisationen mit funktionaler Nähe. In diesem Ansatz wird nicht von der untersuchten Organisation ausgegangen, sondern umgekehrt von den Strukturangleichungsmechanismen (und Ressourcen) ausgehend ein Cluster von ähnlichen Organisationen analytisch zusammengestellt, die die untersuchte Organisation beeinflussen. Die untersuchte Organisation ist also eine Organisation unter vielen (formalen) Organisationen in diesem organisationalen Feld, wobei weitere Umwelten nicht thematisiert werden.

- Scott und Meyer (1991: 117) definieren ihre Konzeption des gesellschaftlichen Sektors als eine gesellschaftliche Umwelt wie folgt:

“A societal sector is defined as (1) a collection of organizations operating in the same domain, as identified by the similarity of their services, products or functions, (2) together with those organizations that critically influ-

---

<sup>198</sup> Ebenso sollte untersucht werden, wo diese Normen in der Organisation integriert werden bzw. wie die Strukturangleichungsmechanismen die Normen in die Organisation einbringen, d.h. wie viele Personen bzw. welche Personengruppen in der Organisation betroffen sind. In diesem Fall handelt es sich aber um Softwareentwickler, die recht homogen in dem Open Source-Projekt auftreten, weswegen diese Problematik vernachlässigt wird.

ence the performance of the focal organizations; for example, major suppliers and costumers, owners and regulators, funding sources and competitors. The adjective societal emphasizes that organizational sectors in modern societies are likely to stretch from local to national or even international actors. The boundaries of societal sectors are defined in functional, not geographical terms.”.

Der Ansatz des gesellschaftlichen Sektors (Societal Sector) geht vom Input bzw. der Leistungsfähigkeit und dem Output einer fokussierten Organisation aus. Er bezieht sich auf Organisationen mit gleichem Produkt oder gleicher Funktion als „Output“, was unschwer als Branche zu erkennen ist. Weitere Organisationen, die zum gesellschaftlichen Sektor zählen, beeinflussen den Input bzw. die Leistungsfähigkeit der fokussierten Organisation, wobei sie nicht nur eine lokale, sondern auch eine nationale sowie internationale Reichweite haben. So beeinflussen z.B. verschiedene Ebenen des Staates wie kommunale, bundesstaatliche und staatliche rechtliche Vorschriften die Organisationen (Powell 1991: 196).

Weiter wird zwischen einem technischen und institutionellen Sektor unterschieden, die jeweils die technischen Prozesse der Organisation kontrollieren und koordinieren oder durch Normen und Regeln beeinflussen. Der technische Sektor bezieht sich vor allem auf materielle bzw. finanzielle Ressourcen und der institutionelle Sektor auf die Versorgung durch Legitimität. Beide Sektoren stellen analytische und keine realen Kategorien dar, wobei aber der technische Sektor eine starke Ähnlichkeit mit der Sanktion und der institutionelle Sektor eine starke Ähnlichkeit mit der Imitation oder Sozialisation (normativem Druck) hat (vgl. Scott, Meyer 1991: 123; Powell 1991). Damit werden ansatzweise Ressourcen zur Lösung des Wahlproblems bei konfligierenden Normen verwendet.

Die Konzeptionen organisationales Feld und gesellschaftlicher Sektor zeigen eine begrenzte Verwendbarkeit für das Open Source-Projekt. Die Problematik der Wahlfreiheit wird bei konfligierenden Normen kaum gelöst (vgl. Walgenbach 1999; Hasse, Krücken 1999). Beim organisationalen Feld gilt für die vielen Organisationen in diesem Cluster weiterhin das Problem, nach welchen Institutionen aus der gesellschaftlichen Umwelt diese sich prioritär ausrichten sollen. Beim gesellschaftlichen Sektor besteht eine Lösung des Problems ansatzweise über den technischen und institutionellen Sektor, wobei von einer untersuchten Organisation ausgegangen wird. Die Unterscheidung in technischen und institutionellen Sektor aber wurde stark diskutiert und mittlerweile relativiert bzw. aufgegeben, da eine klare Abgrenzung zwischen beiden Sektoren kaum möglich war (Walgenbach 1999: 326ff.). Das Konzept des organisationalen Feldes konzentriert sich nicht auf ein Untersuchungsobjekt und wird daher nicht weiterverwendet. Dagegen kann der Blick des gesellschaftlichen Sektors auf eine Organisation für die Untersuchung des KDE-Projekts weiterverwendet werden, wobei von den Erkenntnissen bzgl. dieses Projekts auf alle ähnlichen Projekte geschlossen werden kann.

Gerade bei beiden Konzeptionen steht ein Cluster von Organisationen im Mittelpunkt, was den Blick auf wenige einflussnehmende Umwelten einengt. Besonders die Input-/Output-Orientierung beim gesellschaftlichen Sektor fokussiert zu sehr auf Branchen oder Unternehmen, die sich funktional ähnlich sind. Eine Differenzierung in beeinflussende gesellschaftliche Sphären findet nicht statt, so dass nicht alle Einflüsse aus der gesellschaftlichen Umwelt erfasst werden können, auch wenn sie einen Einfluss haben. Auch die fast ausschließliche Betrachtung von (formalen) Organisationen wird dem KDE-Projekt nicht gerecht (Senge 2005: 96f.). Die gesellschaftliche Umwelt besteht aber nicht nur aus ande-

ren Organisationen, sondern auch aus weiteren nicht-organisationalen Akteuren<sup>199</sup> oder gesellschaftlichen Elementen wie Werte, Rollenmuster, Unternehmenskulturen, Professionsstandards, die in beiden Konzeptionen unbeobachtet bleiben, aber im Open Source-Projekt KDE eine Rolle spielen (vgl. Senge 2005: 92ff.). Die letztgenannten gesellschaftlichen Elemente sind dabei als gesellschaftliche Normen/Institutionen in eine neue Analytik zu integrieren. Beide Konzeptionen können zudem eine soziale Bewegung nicht erfassen, die eine wichtige gesellschaftliche Umgebung des Open Source-Projekts KDE darstellt.

Zudem erfolgt in beiden Konzepten die Imitation von Strukturen vor allem im organisationalen Feld bzw. im gesellschaftlichen Sektor, die Sozialisation und die juristische Sanktion dagegen von außerhalb dieses Feldes oder Sektors. Eine Imitation von Regeln aus anderen Bereichen wird nicht ausgeschlossen, aber selten betrachtet. Untersucht wurde hauptsächlich die Imitation von Strukturen aus dem Nahbereich (d.h. organisationales Feld und gesellschaftlicher Sektor) einer untersuchten Organisation (vgl. Mizuchri, Fein 1999), was auf eine definitionsimaneunte Untersuchungseingrenzung hindeutet. Diese Einschränkung kann nur durch ein neues Konzept für die gesellschaftliche Umwelt vermieden werden.

Einige Gründe sprechen also gegen eine weitere Verwendung der Begriffe organisationales Feld und gesellschaftlicher Sektor in der Analyse des KDE-Projekts, so dass im Weiteren von beeinflussenden Akteuren anstelle von Organisationen gesprochen wird. Statt der Verengung auf den direkten Umkreis eines Untersuchungsobjekts wie bei beiden obigen Konzeptionen wird der gesellschaftliche Bezug eines Untersuchungsobjekts hervorgehoben, weswegen Parsons Gesellschaftstheorie als historischer „Vorläufer“ dieses organisationssoziologischen Theoriekonzepts ein anschlussfähiges Konzept für die weitere Verwendung darstellt (s.a. Senge 2005). Beide Konzepte werden daher verworfen und Parsons Gesellschaftstheorie herangezogen, auch da die soziale Bewegung oder andere Akteure in dieser Theorie verortet werden können. Alle gesellschaftlichen Akteure, die einen Einfluss auf das KDE-Projekt haben können, werden dabei prinzipiell anfangs als gleichwertig betrachtet. Nach Parsons (1972, 1976) werden verschiedene Gesellschaftsbereiche identifiziert, aus denen Normen übernommen werden. In diesen können Kombinationen von Koordinationsformen bestehen und jeweils einen Einfluss auf die Softwareentwicklung im Open Source-Projekt haben.

### Parsons Gesellschaftstheorie und seine Einpassung in den Neo-Institutionalismus

Im Anschluss an Senge (2005: 100ff.) kann Parsons Gesellschaftstheorie in den Neo-Institutionalismus eingepasst werden.<sup>200</sup> Parsons Konzeption bietet sich an, da er erstens Institutionen und ihre Erhaltung wie im Neo-Institutionalismus im Blick hat und zweitens die gesellschaftliche Funktionseinteilung zur Verortung des Open Source-Projekts KDE und zur Identifizierung der beeinflussenden gesellschaftlichen Bereiche herangezogen

---

<sup>199</sup> Ein nicht-organisationaler Akteur ist ein soziales Aggregat, in dem Menschen gemeinsam nach gemeinsamen Normen und Werten handeln und formale sowie informale Organisationen beeinflussen können. Beispielsweise das Schulsystem, der Staat, aber auch die „World Polity“ werden von den Neo-Institutionalisten als beeinflussende Bereiche gesehen (Senge 2005: 97; Hasse, Krücken 1999: 32ff.). Dazu sei angemerkt, dass diese nicht-organisationalen Akteure oder Einflüsse nach ihrer Herkunft aus einer Mikro- und Makroebene unterschieden werden können, was auch für den späteren Einsatz von Parsons Gesellschaftstheorie interessant ist.

<sup>200</sup> Luhmanns Systemtheorie liegt zwar nahe, untersucht aber Kommunikationsmittel anstelle von Handlungen.. Dabei überscheidet sich das Kommunikationsmittel Macht im politischen System mit dem in dieser Arbeit zentralen Begriff Koordinationsform Hierarchie/Macht, was zu Begriffsverwirrungen führen könnte. Außerdem ist seine Theorie eine reine Makrotheorie, in die Organisationen kaum eingepasst werden können. Bei Parsons Theorie bestehen mehr Anschlüsse zum Neo-Institutionalismus als bei Luhmanns Theorie, so dass eine Fusion der anschlussfähigen Theorie eine Weiterentwicklung zu einer Theorie der Koordination anstoßen könnte.

werden kann. Drittens gehen Parsons, der Neo-Institutionalismus und diese Untersuchung von Handlungen aus. Viertens stellt die organisationelle (Meso-)Ebene eine Leerstelle dar, mit der Parsons Theorie in den Neo-Institutionalismus eingepasst werden kann. Fünftens beinhaltet die Parsons'sche Konzeption eine gesellschaftliche, horizontal-funktionale Einteilung in Verbindung mit einer vertikalen Einteilung in Mikro-, Meso- und Makroebene.<sup>201</sup> Außerdem sind die Systeme durch die physische Umwelt territorial begrenzt, wobei die Grenzen je nach Beobachtungsgegenstand und Abstraktionsgrad variieren können (Senge 2005: 95ff.). Dies prädestiniert Parsons Gesellschaftstheorie als ein Element im Theoriekonzept Neo-Institutionalismus für die spätere Analyse von Koordinationsformen im Open Source-Projekt.

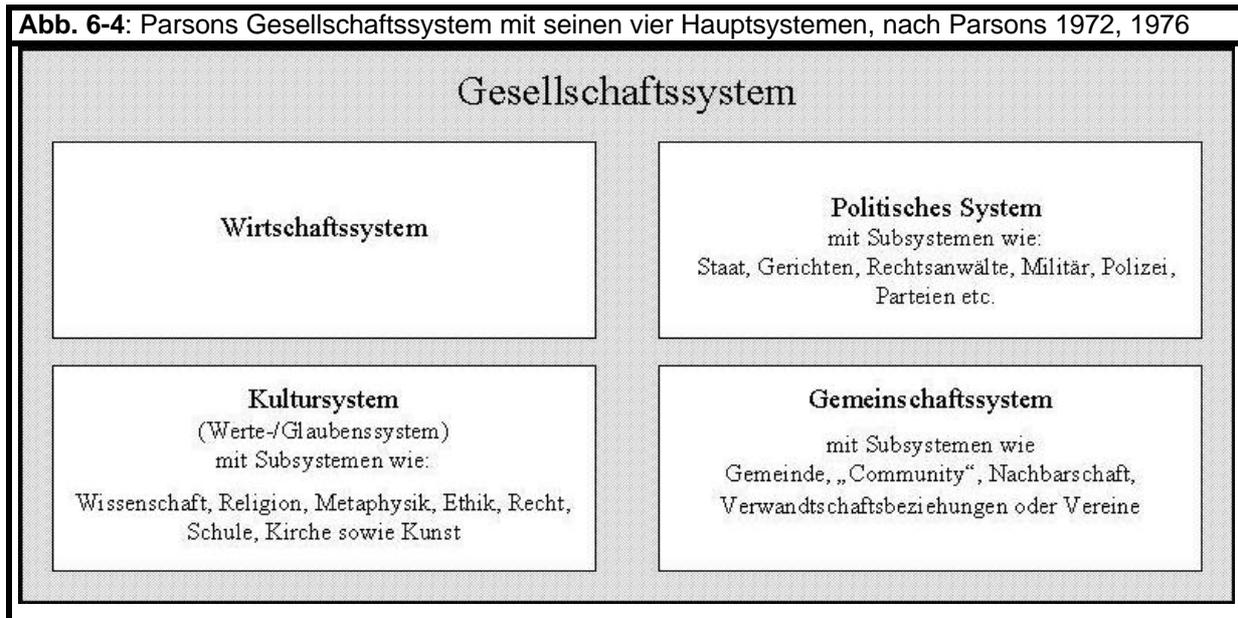
Parsons (1972, 1976) betrachtet moderne, hoch differenzierte Gesellschaftstypen wie die US-amerikanische Gesellschaft, die er analytisch als ein soziales System mit vier gesellschaftlichen (Haupt-)Systemen auffasst. Diese vier Hauptsysteme Wirtschaftssystem, politisches System, Gemeinschaftssystem und Kultursystem können wiederum in Subsysteme aufgeteilt werden<sup>202</sup>, die in Abb. 6-4 dargestellt werden. Dabei gehören soziale Bewegungen als Subsystem in das politische System (vgl. Parsons 1972, 1976; Bühl 1974: 22f.). Die vier gesellschaftlichen Systeme mit ihren Subsystemen können als gesellschaftliche Umwelten bzw. „äußere „Institutionengeber“ die untersuchte Organisation KDE-Projekt als „Institutionennehmer“ beeinflussen (vgl. Senge 2005: 100ff.). Zudem können die Subsysteme für eine detaillierte Feststellung der beeinflussenden gesellschaftlichen Akteure herangezogen werden, wobei Abb. 6-4 bei der Verortung des untersuchten Open Source-Projekts und der beeinflussenden gesellschaftlichen Akteure unterstützt. Dabei werden Ressourcen von den Gesellschaftssystemen analytisch getrennt betrachtet.

---

<sup>201</sup> „Damit schließt dieser [Gesellschafts-]Begriff nicht nur die Organisationen einer Gesellschaft ein, sondern alle beobachtbaren Strukturmuster, wie Rollen, Normen, Werte, Routinen etc. Zudem findet sich bei Parsons die im Neo-Institutionalismus vertikal angelegte Differenzierung der Gesellschaft in eine Mikro-, Meso- und Makroebene. Und es findet sich die im Neo-Institutionalismus mit den Begriffen des ‚organizational field‘ und ‚organizational sector‘ angelegte horizontale Differenzierung der Gesellschaft in funktionale Bereiche, da Parsons die Funktionsbereiche Ökonomie, Politik, Kultur und Gemeinschaft unterscheidet“ (Senge 2005: 99).

<sup>202</sup> Die vier Systeme unterscheiden sich in Ihren Funktionen: Das Wirtschaftssystem ist für die Anpassung an die physische Umwelt, das politische System für eine gemeinsame Zielerreichung durch Organisation von Interessen und Interessenausgleich, das Gemeinschaftssystem für Integration und Erhalt in die gemeinsamen normativen Orientierungen und das Kultursystem für eine Wertestrukturerhaltung durch Erhalt von gemeinsamen Wertvorstellungen zuständig (s. Parsons 1972, 1976 für eine detailliertere Definition).

**Abb. 6-4:** Parsons Gesellschaftssystem mit seinen vier Hauptsystemen, nach Parsons 1972, 1976



Die Gesellschaftsbereiche nach Parsons lassen sich auch deswegen an dieser Stelle für die Studie verwenden, da die Koordinationsformen Kooperation, Hierarchie/Macht und Wettbewerb in den unterschiedlichen Gesellschaftsbereichen vorkommen und das Open Source-Projekt beeinflussen können. Im Wirtschaftssystem können in den Unternehmen Koordinationsformen nachgewiesen werden (s. Kap. 3.1.4., vgl. Boltanski, Chiapello 2003). Das Politiksystem besteht aus einer Mischung von kooperativen, wettbewerblichen (s. Mehrheitswahlen) und hierarchischen Elementen, wobei in der Demokratie diese austariert werden (vgl. Sartori 1992; Heun 1983).<sup>203</sup> Auch im Kultursystem sind Vorstellungen von Kooperation, Hierarchie und Wettbewerb vorhanden (vgl. Thompson et al. 1990). Im Gemeinschaftssystem spielen Koordinationsformen in (Klein-)Gruppen eine Rolle (vgl. Schattenhofer 1997; Rosenstiel et al. 1995; Sader 1991).

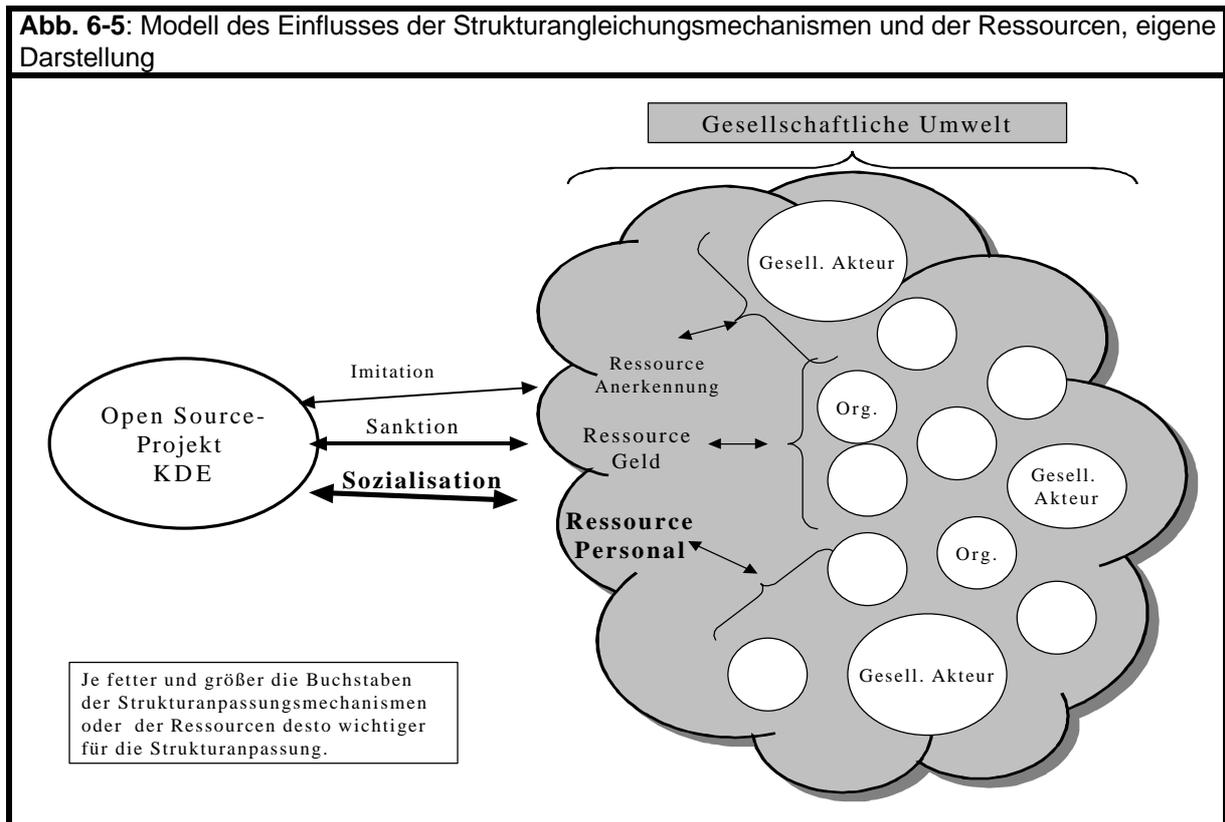
#### 6.2.1.5. Zusammenfassung und Weiterentwicklung der theoretischen Ausführungen

Für eine Analyse der vorgefundenen Koordinationsformen als Institutionen im Open Source-Projekt werden die obigen theoretischen Ausführungen zusammengefasst und ein Modell aus der Theorieinterpretation gebildet. Die untersuchte Organisation ist in dieser Studie das Open Source-Projekt KDE, liegt auf der Mesoebene und steht im Zentrum der Untersuchung. Ausgangspunkt dieser Theorieinterpretation ist, dass das untersuchte Open Source-Projekt KDE und die beeinflussenden gesellschaftlichen Bereiche analytisch zu trennen sind, wobei die untersuchte Organisation in ihren gesellschaftlichen Beziehungen eingebettet ist (vgl. Hasse, Krücken 1999). Von dieser Ausgangsposition werden die Strukturangleichungsmechanismen, Ressourcen und Akteure/Bereiche als analytisch getrennte Kategorien betrachtet, die auf das Open Source-Projekt KDE bezogen werden.

Das untersuchte Open Source-Projekt KDE ist von bestimmten Ressourcen abhängig, die es aus der gesellschaftlichen Umwelt bezieht. Das Projekt erhält Ressourcen aus der gesellschaftlichen Umwelt, indem sie die Legitimität über die Strukturangleichungsmechanismen erhöht. Das untersuchte Open Source-Projekt KDE richtet sich dabei an den Ressourcen aus, die besonders hohe Legitimität von den gesellschaftlichen Akteuren zeigen.

<sup>203</sup> In einer Diktatur, Monarchie o.a. dominiert Hierarchie/Macht zwar stark, aber die anderen Koordinationsformen treten auch auf. Als Beispiel ist die Papstwahl zu sehen, die ein wettbewerbliches Element in der sonst hierarchischen Kirchenorganisation ist.

Die Ressourcen bestimmen im Hintergrund mit, welche Institutionen welcher gesellschaftlicher Akteure bevorzugt institutionalisiert werden (s. Abb. 6-5). Also bestimmen Strukturangleichungsmechanismen und Ressourcen, welche Koordinationsformen aus welchen gesellschaftlichen Systemen und Subsystemen in das Open Source-Projekt KDE importiert, verankert und verkeilt werden. Die wichtigsten Strukturangleichungsmechanismen und Ressourcen bestimmen dann, welche Institutionen aus der gesellschaftlichen Umwelt dominieren und welche nachgeordnet sind. Damit können die Ausprägungen und der Stellenwert der Koordinationsformen analysiert werden.



Die Wichtigkeit der Strukturangleichungsmechanismen und Ressourcen werden jeweils auf einem anderen Weg festgestellt. Dabei lösen wichtige Strukturangleichungsmechanismen, die für die Legitimitätserhöhung der Organisation sorgen, und wichtige Ressourcen das Problem der Wahlfreiheit bei konfligierenden Normen. Eine Institution ist besonders stark institutionalisiert, wenn eine Institution nicht durch eine andere Institution aufgehoben bzw. deinstitutionalisiert wird, d.h. wenn für eine Situation keine konfligierenden Institutionen bestehen (Zucker 1991: 728). Damit steht das untersuchte Open Source-Projekt in einem Gleichgewicht mit den Einflüssen aus der gesellschaftlichen Umwelt, d.h. ein hoher Institutionalierungsgrad besteht, wenn sich die Institutionen aus den gesellschaftlichen Umwelten und die Strukturangleichungsmechanismen ergänzen. Eine Institution ist daher stärker institutionalisiert und damit bedeutender, je mehr der drei Strukturangleichungsmechanismen aus den verschiedenen Gesellschaftsbereichen diese Institution unterstützen und je mehr ähnliche Institutionen aus verschiedenen Gesellschaftsbereichen einwirken. Ständige Institutionalierungen und De-Institutionalisierungen werden durch verschieden stark institutionalisierende Strukturangleichungsmechanismen abgedeckt. Die drei Strukturangleichungsmechanismen verkeilen bzw. bestärken die Institutionen also unterschiedlich in dem Institutionengefüge des untersuchten Open Source-Projekts. Die

Stärke der Institutionalisierung durch einen Mechanismus ist abhängig von der Resistenz gegenüber individuellen und kurzfristigen Änderungen. Koordinationsformen, die in die organisationale Struktur übernommen werden, sind über die kaum zu ändernde Sozialisation stärker institutionalisiert und damit bedeutender als über die Politik veränderbare Sanktion und am wenigsten institutionalisiert über die kurzfristig einsetzbare Imitation. Eine Vertragseinhaltung wird durch das juristische Subsystem gesichert.

Die Ressourcen können auch in eine Rangordnung der Wichtigkeit für die gesellschaftlichen Akteure als Legitimationsgeber eingeteilt werden, wobei die Wichtigkeit durch die Stärke der Beteiligung angezeigt wird. Eine besondere Beteiligung ist bei Personal/Arbeitszeit und geschriebener Software bzw. Informationen vorhanden, da sich der Legitimationsgeber vollständig einbringt. Finanzielle und materielle Ressourcen wie Hardware stellen nur eine Verhaltenseinschränkung mit geringer Beteiligung für den gesellschaftlichen Akteur dar und haben daher nur eine mittlere Bedeutung. Die geringste Bedeutung hat Anerkennung, da sich der Legitimationsgeber kaum beteiligen muss.

Folglich sind diejenigen Normen am stärksten institutionalisiert und am bedeutendsten, die mit der wichtigsten Ressource Personal zusammenhängen und über Sozialisation kaum veränderbar sind.<sup>204</sup> Institutionen können über alle drei oder einzelne Strukturangleichungsmechanismen der Legitimitätserhöhung aus dem gesellschaftlichem Bereich in das untersuchte Open Source-Projekt KDE übertragen werden. Ähnliches gilt auch für Ressourcen. Ein Gesellschaftsbereich kann nur eine Ressource, aber auch mehrere Ressourcen für das untersuchte Open Source-Projekt KDE bereitstellen.

Folglich werden Ausprägungen und Stellenwert der Koordinationsformen im Open Source-Projekt KDE einerseits über die organisationalen Ressourceninputs von Akteuren aus verschiedenen gesellschaftlichen Bereichen, andererseits über die Strukturangleichungsmechanismen analysiert, die die Normen aus der gesellschaftlichen Umwelt in das Open Source-Projekt KDE übertragen. In beiden Fällen spielen die Akteure aus der gesellschaftlichen Umwelt eine Rolle, aus denen die Institutionen importiert werden. Nach Parsons sind gesellschaftliche Bereiche das Wirtschafts-, Politik-, Kultur- und Gemeinschaftssystem, die sich weiter in Subsysteme ausdifferenzieren (s. Abb. 6-4). Das untersuchte Open Source-Projekt KDE liegt auf der Mesoebene zwischen den beeinflussenden gesellschaftlichen Systemen auf der Makroebene (Kultur-, Politik-, Wirtschaftssystem) und der Mikroebene (Gemeinschaftssystem). Mittels Strukturangleichungsmechanismen und Ressourcenbezug können die Akteure innerhalb der Teilsysteme bzw. gesellschaftlichen Bereiche festgestellt werden, aus denen die Normen kommen. Dabei kann das untersuchte Open Source-Projekt KDE auch direkten Kontakt mit diesen Akteuren aus diesen Gesellschaftsbereichen haben (z.B. mit anderen Organisationen: Walgenbach 2002: 174f.). Zusätzlich kann das untersuchte KDE-Projekt einem der Gesellschaftsbereiche zugerechnet werden, da es nicht im „luftleeren“ Raum schwebt, sondern in die gesellschaftlichen Bezüge eingebunden ist. Der gesellschaftliche Bereich, dem die untersuchte Organisation angehört, hat womöglich einen besonderen Einfluss auf diese (vgl. Garud, Kumaraswamy 1995; Lant, Baum 1995).

Wird Parsons Gesellschaftstheorie in das Modell integriert, fehlt noch der Bezug zu anderen ähnlichen Organisationen des untersuchten Open Source-Projekts KDE. Diese ähnlichen Organisationen werden als organisationales Feld oder gesellschaftlicher Sektor bezeichnet (s.o.). Da aber beide Konzeptionen Schwierigkeiten bei der Konzeptionalisierung

---

<sup>204</sup> Sozialisierte Normen mit organisationszentralen Ressourcen sind wahrscheinlich eher in der informalen institutionellen Aktivitätsstruktur, dagegen imitierte und sanktionierte Normen mit unwichtigeren Ressourcen wahrscheinlich eher in der Formalstruktur bzw. als „Talk“ anzutreffen.

für das KDE-Projekt haben, wird auf den Begriff der „Bezugsgruppen“ in Anlehnung an den Begriff von individuellen Bezugsgruppen in der Sozialpsychologie zurückgegriffen. Dabei „fühlt“ sich die untersuchte Organisation Open Source-Projekt KDE zu einer Ansammlung von ähnlichen Gruppen, Organisationen oder Akteuren mit bestimmten Bezugsnormen zugehörig, kann aber auch vereinzelt Kontakte zu Teilen dieser Organisationen bzw. Akteure haben (vgl. Gukenbiehl 1999; Garud, Kumaraswamy 1995; Lant, Baum 1995).<sup>205</sup>

Da das untersuchte Open Source-Projekt KDE in ausgeglichenen gesellschaftlichen Bezügen steht, setzt sich die Bezugsgruppe aus sehr ähnlichen Organisationen oder Akteuren zusammen. Diese Organisationen oder Akteure kommen aus dem gleichen gesellschaftlichen Bereich, in dem die untersuchte Organisation verortet werden kann. Die Ähnlichkeit der Organisationen könnte dabei auf das gleiche Ziel bzw. gleiche (Dienstleistungs-)Produkt, das gleiche gesellschaftliche Subsystem und räumliche Nähe zurückgeführt werden. Außerdem bestehen u.U. vermehrt direkte Kontakte zwischen diesen Organisationen oder Akteuren. Diese Organisationen bzw. Akteure der Bezugsgruppe benötigen zudem gleiche Ressourcen gleicher Umwelten, unterscheiden sich aber durch eine leicht unterschiedliche Abhängigkeit von den Ressourcen und Umwelten. Damit gibt es bei den Open Source-Projekten als Organisationscluster einerseits eine Mehrzahl an gleichen Institutionen, aber auch einige Unterschiede (s. Kap. 5.1.), wie es auch Powell (1991, 195) und Scott (1991: 172) feststellten. Für diese Unterschiede kann diese Theorieinterpretation durch die unterschiedliche Institutionalisierungsstärke der Strukturangleichungsmechanismen und der Ressourcen mögliche Erklärungen liefern.<sup>206</sup>

### **6.2.2. Analyse der Koordinationsformen im Open Source-Projekt KDE**

Nach der im vorigen Unterkapitel zusammengefassten theoretischen Grundlage wird im Folgenden Ausprägungen und Stellenwert der vorgefundenen Koordinationsformen im Open Source-Projekt KDE analysiert und Vermutungen über die Einflüsse der Akteure aus den Gesellschaftsbereichen auf diese angestellt. Da keine empirischen Daten vorliegen, wird auf bestehende empirische Literatur zu den Akteuren der Gesellschaftsbereiche zurückgegriffen.

Diese zweite Stufe der Analyse unterteilt sich in einen ersten Zwischenschritt, der im folgenden Unterkapitel 6.2.2.1. die beeinflussenden Gesellschaftsbereiche in den Mittelpunkt stellt. Das Open Source-Projekt KDE kann dabei Koordinationsformen als Institutionen mit ihrem jeweiligen Stellenwert aus spezifischen Gesellschaftsbereichen übernehmen. Wichtige Strukturangleichungsmechanismen und Ressourcen können die Gesellschaftsbereiche feststellen, aus denen die Koordinationsformen kommen. Damit kann bestimmt werden, von welchen gesellschaftlichen Akteuren die Koordinationsformen als Institutionen kommen, die in den relativ neu entstandenen KDE-Projekt vorgefunden werden können. Dies kann durch den vorliegenden Abstraktionsgrad die Literatur nachvollziehbar dargestellt werden.

Im darauffolgenden Unterkapitel 6.2.2.2. als zweitem Zwischenschritt wird die vermutete Übernahme der Koordinationsformen von den Akteuren der gesellschaftlichen Bereiche in den Teilprozessen Eintritt/Rekrutierung, Arbeitsplanung (Zieldefinition, Arbeitsverteilung

---

<sup>205</sup> Die technische Entwicklung der Computerchips beispielsweise ist durch Bezugsgruppen bestimmt (vgl. Garud und Kumaraswamy 1995). Diese Bezugsgruppen bestehen z.B. auch in der Hotelbranche in Manhattan, New York, durch die Einschätzung von Managern über ihre Konkurrenten (vgl. Lant, Baum 1995).

<sup>206</sup> Mögliche Gründe für minimale Unterschiede zwischen den Open Source-Projekten werden im Kapitel 6.2.2.1. unter der Überschrift „Andere Open Source-Projekte als Bezugsgruppe“ vorgestellt.

von Arbeitspaketen mit detaillierter Entwicklungsplanung, Arbeitsverteilung von Posten), Leistungserstellung mit detaillierter Entwicklungsplanung und Arbeitszusammenführung des Open Source-Projekts KDE betrachtet. Dabei wird analysiert, welche Koordinationsformenkombinationen aus den Akteuren der Gesellschaftsbereiche den jeweiligen Teilprozess beeinflussen. Den Einfluss auf die Ausprägung und Bedeutung der Koordinationsformen ist aber durch den Rückgriff auf die Literatur eingeschränkt, da erstens in der Literatur keine Teilprozesse bzw. kein Arbeitsablauf für die gesellschaftlichen Akteure thematisiert werden, zweitens Koordinationsformen nicht eindeutig dargestellt werden und drittens der Detaillierungsgrad der Literatur gering ist. Ein empirisches Vorgehen nach dieser Theorieinterpretation wird wahrscheinlich eher zu einem konkreten Ergebnis führen. Trotzdem können durch das Heranziehen der Literatur ein paar empirisch überprüfbare Vermutungen entwickelt werden.

#### *6.2.2.1. Einfluss der Gesellschaftsbereiche auf das KDE-Projekt*

In diesem Unterkapitel können aus einer Perspektive des Open Source-Projekts KDE über die Nutzung von Strukturangleichungsmechanismen und Ressourcen die relevanten Akteure der gesellschaftlichen Bereiche und ihr spezifischer Einfluss bestimmt werden. Die beeinflussenden Akteure und ihr Einfluss werden dabei analysiert, indem nacheinander die Gesellschaftssysteme bzw. ihre Subsysteme fokussiert werden. Von den Akteuren aus diesen Gesellschaftsbereichen werden die Koordinationsformenkombinationen ins Open Source-Projekt übernommen. Damit lassen sich Vermutungen über die Zusammenhänge aufstellen. Die Institutionalisierungsstärke der Strukturangleichungsmechanismen und die Wichtigkeit der verschiedenen Ressourcen sind aus der Theorie bekannt, wobei bei der Bewertung die Strukturangleichungsmechanismen im Vordergrund und die Ressourcen im Hintergrund stehen. Bei der Analyse der Zusammenhänge zwischen dem untersuchten KDE-Projekt und dem jeweiligen gesellschaftlichen Akteur dieser Systeme wird folgendermaßen vorgegangen:

In einem ersten Abschnitt wird anhand von Strukturangleichungsmechanismen und Ressourcen festgestellt, warum der betrachtete gesellschaftliche Akteur eines der Teil- und Subsysteme Einfluss auf das Open Source-Projekt KDE hat. Theoretisch können alle drei Strukturangleichungsmechanismen oder alle Ressourcen relevant für die Beziehung zwischen KDE-Projekt und gesellschaftlichem Akteur bzw. Bereich sein, doch wirken nicht alle Strukturangleichungsmechanismen oder Ressourcen in jeder dieser Beziehungen. Es werden daher weitestgehend die zentralen Strukturangleichungsmechanismen oder Ressourcen dargestellt, über die die Koordinationsformen übernommen und verkehrt werden. Der zentrale Strukturangleichungsmechanismus Sozialisation und die wichtigste Ressource Personal bestimmen die wichtigsten Akteure der Gesellschaftsbereiche und Subsysteme für das Open Source-Projekt KDE. Sanktion, Imitation und weitere Ressourcen geben nachgeordnete Akteure der Gesellschaftsbereiche an, die auch einen Einfluss haben.

Im zweiten Abschnitt werden die Koordinationsformen vorgestellt, die in dem einflussreichen Akteur bestehen. Als zentraler Ansatz werden zum einen die Koordinationsformen in den Akteuren der Gesellschaftsbereiche, d.h. in dem Institutionengeflecht der Akteure, betrachtet. Zum anderen werden ggf. die Koordinationsformen in den Beziehungen zwischen gleichen Akteuren in einem Gesellschaftsbereich analysiert, wenn diese auftreten und auch einen Einfluss auf die Institutionen der Akteure, aber auch des untersuchten KDE-Projekts haben können. Ohne Probleme können dabei ähnliche Ausprägungen der Koordinationsformen im Institutionengeflecht des KDE-Projekts und relevanter Akteure des Gesellschaftsbereichs festgestellt werden. Der unterschiedliche Stellenwert der Koordinationsformen im Institutionengeflecht des relevanten Akteurs eines Gesellschaftsbereichs kann nur manchmal aus der Literatur entnommen werden, wenn Bezüge und Be-

deutungsunterschiede zwischen den Institutionen benannt werden. Die Übernahme einzelner Koordinationsformen über einen Strukturangleichungsmechanismus sind auf diesem Abstraktionsgrad kaum nachzuvollziehen. Stattdessen wird betrachtet, wieviel gleiche Koordinationsformen aus verschiedenen gesellschaftlichen Akteuren eine Koordinationsform im KDE-Projekt stark institutionalisieren.<sup>207</sup>

Diese Bewertung über Strukturangleichungsmechanismus und Ressourcen aus dem jeweiligen ersten Abschnitt spiegelt sich im Folgenden in der Reihenfolge der Darstellung der Gesellschaftsbereiche mit den relevanten Akteuren wider, d.h. die Gesellschaftsbereiche für das Open Source-Projekt KDE werden nach ihrer Wichtigkeit nacheinander behandelt. Den Anfang macht das Gemeinschaftssystem, in dem das untersuchte Open Source-Projekt als freiwillige, selbstorganisierte Organisation verortet wird. Danach werden nacheinander die Akteure des Kultur-, Politik- und schließlich des Wirtschaftssystems in absteigender Wichtigkeit untersucht. Dabei wird auch der Kontakt zwischen Open Source-Projekt und Akteur sowie die Selbstzuschreibung zu einer Bezugsgruppe von Akteuren mit bedacht (vgl. Walgenbach 2002: 175). Die Bezugsgruppe besteht dabei aus anderen Open Source-Projekten.

Nachdem alle relevanten gesellschaftlichen Bereiche mit ihren Akteuren untersucht wurden, wird am Schluss das Zusammentreffen der verschiedenen Koordinationsformenkombinationen verschiedener Akteure aus verschiedenen Gesellschaftsbereichen diskutiert. Dabei wird untersucht, inwieweit sich die Koordinationsformenkombinationen verschiedener Akteure nicht widersprechen, also gegenseitig verkeilen. Da das KDE-Projekt im Gleichgewicht mit seiner gesellschaftlichen Umgebung steht, dürften kaum konfligierende Normen eine Rolle spielen; der Blick ist nur auf die verkeilenden Mechanismen ausgerichtet.<sup>208</sup> Eine Koordinationsform ist stark institutionalisiert, wenn gleiche bzw. ähnliche Ausprägungen der Koordinationsform aus verschiedenen Gesellschaftsbereichen diese über den wichtigsten Mechanismus Sozialisation im KDE-Projekt verkeilen. Da Kooperation hoch bedeutend ist, ist daher zu erwarten, dass Kooperation in vielen gesellschaftlichen Akteuren existiert und über Sozialisation im KDE-Projekt institutionalisiert wird.

Im Folgenden beginnt die Analyse mit der selbstorganisierten Gruppen aus dem Gemeinschaftssystem.

### Gemeinschaftssystem mit selbstorganisierten Gruppen

Das Open Source-Projekt KDE kann zum Gemeinschaftssystem gezählt werden, da die freiwilligen Mitarbeiter Privatleute sind, auf die das Open Source-Projekt KDE angewiesen ist und die für das Projekt wichtigste Ressource Personal darstellen. Die Privatleute gehören zum Gemeinschaftssystem, weil sie in Familien, Freundschaftsbeziehungen (Cliques) oder regionalen Gemeinschaften sozialisiert wurden. Der Bezug zur Gemeinschaft zeigt sich auch darin, dass einige der befragten KDE-Mitglieder einer ehrenamtlichen Tätigkeit weiterhin nachgehen oder nachgegangen sind (s. Kap. 5.2.). Open Source-Projekte gehören zudem zu den virtuellen Gruppen, die allgemein als Gemeinschaften verstanden werden (Haythornthwaite et al. 2000: 374ff.; Heintz 2000b).

---

<sup>207</sup> Zur Komplexitätsreduktion werden die indirekten Einflüsse zwischen den gesellschaftlichen Akteuren nicht untersucht, z.B. zwischen selbstorganisierter Gruppe und sozialer Bewegung. Da eine soziale Bewegung auch auf selbstorganisierten Gruppen basiert, kann hier eine Selbstverstärkung bestehen. Es wird daher zwischen dem Einfluss beider Akteure streng getrennt.

<sup>208</sup> Falls konfligierende Institutionen auftreten, wird untersucht, warum diese keinen Einfluss auf das KDE-Projekt haben.

In dem zivilgesellschaftlichen und lebensweltlichen Bereich des Gemeinschaftssystems bestehen selbstorganisierte, freiwillige, informale, sekundäre Gruppen, wie Selbsthilfegruppen, Gleichaltrigengruppen (Cliquen) oder Nachbarschafts- oder Stadtteilinitiativen (vgl. Schäfers 1999; Schattenhofer 1992: 16f.; Bernart 1999; Machwirth 1999). Dazu gehören auch überwiegend männliche Hobbygruppen mit Bastlermentalität, wie selbstorganisierte Radiobastler, Radioamateure, Auto-/Motorradbastler, die als Vorläufer der Open Source-Projekte als selbstorganisierte Gruppe verstanden werden können (vgl. Schmid 1997).

Das Open Source-Projekt KDE selbst kann als eine selbstorganisierte Gruppe gesehen werden, da es selbstorganisiert zum Zweck der freiwilligen Softwareentwicklung von autonomen Privatpersonen gegründet wurde (vgl. Schattenhofer 1992). Außerdem kann das KDE-Projekt als eine Gruppe bezeichnet werden, da die Definitionsmerkmale einer Gruppe anzutreffen sind: Im KDE-Projekt besteht ein „Wir-Gefühl“ der Gruppenzugehörigkeit und des –zusammenhalts über Identifikationssymbole (z.B. KDE-Logo<sup>209</sup>) und verschiedene Kommunikationskanäle<sup>210</sup>, ein gemeinsames Gruppenziel der Softwareentwicklung, ein System gemeinsamer Normen/Werte der Netiquette und eine Rollendifferenzierung mit Posten und einer strukturierten Arbeitsteilung zwischen den Beteiligten (vgl. Kap. 5). Weiterhin erfolgt eine ständige und fortgesetzte Kommunikation im Projekt, wobei eine Gruppengrenze über Zugangsbarrieren ins Versionsmanagementsystem definiert ist.<sup>211</sup> Das Open Source-Projekt ist für die Mitarbeiter eine Mitgliedsgruppe, für viele Nutzer aber eine Bezugsgruppe ohne Mitgliedschaftsstatus.

Das Projekt gehört in seiner Gesamtheit den Großgruppen an, besteht aber in seinem modularen Aufbau der IT-Infrastruktur aus ineinander gestaffelten Gruppen bzw. Kleingruppen mit unter 25 Personen (vgl. Schäfers 1999; Claessens 1999; Döring 2003: 492; Sader 1991: 37ff.).<sup>212</sup> Durch diese Modularisierung der Großgruppe in Kleingruppen können einerseits die Erkenntnisse der realweltlichen Kleingruppenforschung herangezogen werden. Andererseits ist das KDE-Projekt eine Großgruppe, mit mehr oder weniger informalen oder formalen Institutionen. Erkenntnisse über realweltliche, selbstorganisierte, informale Großgruppen bestehen in der Literatur im Wesentlichen abgesehen vom Begriff der Masse und sozialen Bewegungen kaum<sup>213</sup>. Da soziale Bewegungen als informale

---

<sup>209</sup> Die Identifikation mit der Gruppe erfolgt über spezifische Symbole wie das KDE-Zahnrad, den Drachen des Webbrowsers Konqueror und der Nutzung des Anfangsbuchstabens K für Softwarenamen als Running Gag (s. Kap. 5).

<sup>210</sup> Sind virtuelle Gruppen oftmals auf ein spezifisches Medium wie nur ein Forum, eine Usenet-Liste, ein MUD/MOO oder eine Mailingliste bezogen, nutzt das Open Source-Projekt verschiedene diverse Kommunikationsmedien mit unterschiedlichem Intimitätsgrad (s. Kap. 5.3.). Ein hohes Zugehörigkeitsgefühl besteht gerade in Chats, in denen neben den eher sachlichen Mailinglisten eher über persönliches kommuniziert wird.

<sup>211</sup> Döring (2003) sieht noch Kollaboration bzw. wechselseitige Unterstützung der Mitglieder als Definitionskriterium (collaboration), diese werden aber in den anderen Definitionen nicht verwendet (vgl. Sader 1991; Schäfers 1999).

<sup>212</sup> Bei einem Anwachsen der Kleingruppe, was beim KDE-Projekt der Fall ist, entstand also eine Großgruppe mit unterschiedlichen Kleingruppen in horizontaler und vertikaler Arbeitsteilung. Einerseits bestehen Gruppen „erster Ordnung“, die die Arbeit erledigen und eigene Subinteressen verfolgen, sowie Gruppen „zweiter Ordnung“ aus Vertretern von Untergruppen bzw. den Aktiven, die die „Verwaltung“/Planungsvorbereitungen und Außendarstellung der Gruppe übernehmen. Andererseits bildeten sich neue oder untergeordnete Funktionen, die sich beim KDE-Projekt in verschiedenen Tätigkeiten (Übersetzung, Dokumentierung, Konferenzvorbereitung etc.) widerspiegeln (vgl. Claessens 1983: 492, 1999; Rauch 1983).

<sup>213</sup> Informale Großgruppen können mit dem Begriff der Masse in Zusammenhang gebracht werden. Da aber Massen eher chaotisch strukturiert sind und eher durch Emotionen getrieben werden, also Institutionen keine Rolle spielen, wird auf eine Untersuchung verzichtet.

Großgruppen aber zum politischen System gezählt werden, werden diese hier ausgeklammert.

Das Open Source-Projekt KDE als selbstorganisierte (Klein-)Gruppe kann sich auf den sozialisierten Institutionen der realweltlichen (selbstorganisierten) Gruppen stützen, da das KDE-Projekt als virtuelle Gruppe in der Biografie der Mitglieder nach einer Sozialisation in solchen realweltlichen (selbstorganisierten) Gruppen auftritt. Außerdem können realweltliche Gruppen in formale und informale Gruppen unterschieden werden, in die Individuen sozialisiert werden. Kinder und Jugendliche sozialisieren sich z.B. gegenseitig in informellen Peer Groups (Gruppe von Gleichaltrigen), aber auch in informalen Sportgruppen in formalen Sportvereinen. Auch in der Schule als Formalstruktur<sup>214</sup> bestehen Schulklassen, in denen gleichzeitig informal gebildete, freiwillige Freundesgruppen existieren (vgl. Herlyn 1999; Machwirth 1999; Setzen 1999, Jütting et al. 2003). Selbstorganisierte, informale und formale Gruppen existieren also zugleich, wobei formale Gruppen oftmals den äußeren Rahmen für informale Gruppen abgeben. Die Formalstruktur bzw. das Institutionengeflecht z.B. eines Sportvereins wird dadurch nicht automatisch sozialisiert, da die formalen Sitzungen meist unabhängig von den Sportaktivitäten eines Vereins erfolgen. Viele Mitglieder nehmen an diesen Sitzungen nicht oder selten teil. Trotzdem kann eine formale Vereinsstruktur mit seiner Satzung Einfluss auf die Aktivitätsstruktur nehmen. Dies ist im KDE-Projekt nicht der Fall, obwohl ein Verein (KDE e.V.) gegründet wurde. Durch das Strukturprinzip der Selbstorganisation ähneln sich informale (Klein-)Gruppen stark, dagegen unterscheiden sich die formalen (Groß-)Gruppen des Vereins in ihren Institutionen, werden aber durch das Vereinsgesetz zu einem Minimalmaß an Kongruenz gezwungen. Aus diesem zusätzlichen Grund wird die Formalstruktur des Vereins und formale Großgruppen unter dem Politiksystem und seinem juristischen Subsystem betrachtet. Auf Erkenntnisse über realweltliche formale Großgruppen wird dann zurückgegriffen, wenn es um Gruppen zweiter Ordnung<sup>215</sup> geht (vgl. Schäfers 1999: 23) und diese auch im KDE-Projekt auftreten. An dieser Stelle wird daher gezeigt, wie selbstorganisierte, informale Gruppen die institutionelle Binnenstrukturierung des KDE-Projekts bestimmen.

Zwar unterscheidet sich das KDE-Projekt von realweltlichen Gruppen durch die überwiegende Nutzung der IuK-Technik bzw. die seltenen Zusammentreffen und die rein textbasierte Kommunikationsart, doch bestehen überwiegend Gemeinsamkeiten. Virtuellen, aber auch nicht-virtuellen selbstorganisierten Gruppenarten ist eine starke Themenzentriertheit bzw. ein gemeinsames Interesse sowie gemeinsame Aktivitäten gemeinsam. In Gruppen, auch virtuellen Gruppen, treten zwei Arten des gemeinsamen Interesses auf, die sich auf die gemeinsame Arbeit bzw. Leistungserstellung oder auf Geselligkeit mit sozio-emotionalem Austausch (Gemeinschaftsaspekt) beziehen.<sup>216</sup> Bei manchen, auch virtuellen Gruppen, ist der Gemeinschaftsaspekt stärker ausgeprägt, bei anderen wie beim Open Source-Projekt KDE dagegen steht die gemeinsame Leistungserstellung im Vordergrund und der Gemeinschaftsaspekt spielt neben der Arbeit wie Softwareentwicklung eine geringere Rolle (vgl. Götzenburger, Löger 2000: 258; Wellman 2000: 150; Levine 2001; Döring 2003: 490ff.; Thiedecke 2000a, b; Kollock 1999). Die virtuellen Gruppen können zwar eigene Qualitäten durch die Möglichkeit der Bildung von virtuellen Identitäten entfalten, sind aber in vielen Fällen stark an realweltlichen Gruppen mit realen Identitäten orientiert

---

<sup>214</sup> Die Schule gehört eigentlich zum Kultursystem und wird hier nur zur Illustration verwendet.

<sup>215</sup> Diese Begrifflichkeit stammt aus der Systemtheorie und beschreibt eine Gruppe, die auf einer abstrakteren Ebene die Interessen der Basisgruppen integriert, aber selbst eine Gruppe darstellt (vgl. Claessens 1999).

<sup>216</sup> Die Themenzentriertheit bei virtuellen Gruppen äußert sich z.B. über den alleinigen Bezug auf das Spielen in virtuellen Umgebungen, den Austausch von Musikaufnahmen, die Kommunikation über ein spezifisches emotionales Problem, aber auch auf Softwareentwicklung (Schattenhofer 1992: 128).

(Thiedecke 2000a: 8f., 13). Dies gilt auch für das Open Source-Projekt KDE, bei dem sich einzelne Kleingruppen hin und wieder, ein Großteil des KDE-Projekts jedoch an der jährlichen Konferenz persönlich treffen. Virtuelle Gruppen bestehen aus realweltlichen Gruppen mit zusätzlicher virtueller Kommunikation oder sie basieren auf Erfahrungen realweltlicher Gruppen wie beim KDE-Projekt. Es wird daher hauptsächlich Literatur zu selbstorganisierten, realweltlichen, (Klein-)Gruppen und, wenn angebracht, auch Literatur zu virtuellen Gruppen herangezogen.

In den freiwilligen, selbstorganisierten, informalen Gruppen, aber auch in virtuellen Gruppen sind vor allem kooperative, aber auch hierarchische Elemente anzutreffen, wobei Wettbewerb als Innergruppenkonflikt thematisiert wird (vgl. Sader 1991: 136ff.; Grunwald, Lilge 1981; Levine, Moreland 1995; Kollock 1999; Levine 2001; Götzenburger, Löger 2000; Hoffmann 2000; Döring 2003: 490ff.). Diese Koordinationsformen wurden in diesen Gruppen sozialisiert und können beim KDE-Projekt angewendet werden.

- Basis einer Gruppe wie dem Open Source-Projekt KDE ist eine reziproke und eine bedeutende gruppenorientierte *Kooperation* mit starker Vertrauenskomponente, was sich in einem gemeinsamen Interesse als Grundkonsens widerspiegelt. Dabei können Gefühle wie Liebe, Treue, Solidarität und Dankbarkeit mit hoher Emotionalität und freundlicher Kommunikation den Grundkonsens einer selbstorganisierten Gruppe unterstützen (s.o.) und zur Stabilität einer Gruppe wie dem KDE-Projekt beitragen. Außerdem unterstützt in selbstorganisierten, informalen Gruppen eine Norm der Gleichstellung die gegenseitige Hilfe (Neidhardt 1999: 148f.; Levine, Moreland 1995: 449; Schattenhofer 1992; Bernart 1999: 297f.). Beispiele für Gruppen mit dominierender Kooperation sind neben dem KDE-Projekt informale Gruppen des Freizeitsports und Selbsthilfegruppen (vgl. Setzen 1999; Bernart 1999). Bei Selbsthilfegruppen mit ihrem therapeutischen Charakter werden einige Elemente der selbstorganisierten Gruppen wie Selbstbestimmung, Offenheit, Hoffnung und Solidarisierung in Gesprächen besonders betont (Bernart 1999: 297f.), die auch im KDE-Projekt vorgefunden werden können. Das Open Source-Projekt KDE ist zwar keine Selbsthilfegruppe im therapeutischen Sinne, kann aber als eine Art Selbsthilfe gegenüber proprietärer und kommerzieller Software gesehen werden und bezieht sich dabei auf die soziale Open Source-Bewegung.

Das themenzentrierte Diskutieren mit einem „zwanglosen Zwang eines besseren Arguments“ steht bei Entscheidungen im Vordergrund, wobei Konsens angestrebt wird (Sader 1991: 215, 229ff.; Paris 1998: 41; Habermas 1997). Die Themenzentriertheit zeigt sich auch darin, dass sich in selbstorganisierten Gruppen wie Open Source-Projekten Freundschaften zwischen einzelnen Beteiligten bilden können, die sich u.U. zu Freundeskreisen weiterentwickeln können. Die privaten Beziehungen werden jedoch von der Gruppe abgelöst, was besonders bei Selbsthilfegruppen, aber auch beim KDE-Projekt beobachtet werden kann (Schattenhofer 1992: 116ff., 129; vgl. Bernart 1999).

- Es ist weiterhin aus der Kleingruppenforschung bekannt, dass in freiwilligen Gruppen eine aufgabenorientierte und eine sozio-emotionale Führungsrolle (*Hierarchie/Macht*) nachgeordnet neben der Kooperation bestehen, die mehr oder weniger getrennt auf ein oder zwei Personen aufgeteilt sind (vgl. Sader 1991). Diese Führungsrollen sind eine spezifisch informale Führerschaft bzw. Leitung, da die Positionen nicht festgeschrieben wurden und keine offizielle Übertragung von Posten erfolgt. Bei virtuellen Gruppen wie dem KDE-Projekt entstehen neben den informalen Führerschaften, Leitungsfunktionen und Reputationshierarchien auch eher informale technische Posten mit Zugriff auf die IuK-technische Infrastruktur und dem Einsatz von IuK-technischen Sanktionen (vgl. Schatten-

hofer 1992: 154f.; Thiedecke 2000b; Döring, Schestag 2000; Götzenburger, Löger 2000; Butler et al. 2002).<sup>217</sup>

Es entsteht ein innerer Kern in einer Gruppe wie dem Open Source-Projekt KDE, der mehr Verantwortung übernimmt und Entscheidungen herbeiführt. Der innere Kern bildet sich aus Engagierten, Erfahrenen und Gruppengrüdern, wobei die Reputation und hohe Beteiligung bzw. Aktivität eine Rolle spielt und deswegen als „Reputationshierarchie“ bzw. „meritokratische Oligarchie“ bezeichnet werden kann. Im Unterschied zur formalen Positionshierarchie werden die Aufgaben und ihre Bewältigung nicht festgelegt, auch um die fiktive Gleichheit aller Beteiligten weiterhin zu ermöglichen (vgl. Götzenburger, Löger 2000; Schattenhofer 1992: 134ff.; Sader 1991: 288ff.). Eine Sanktionierung von Normbrüchen erfolgt eher über einen Zweifel an der moralischen Integrität eines Gruppenmitglieds, was auch in den textbasierten Sanktionen im KDE-Projekt angesprochen wird (Neidhardt 1999: 148f.).

- Konflikte als *Wettbewerb* in freiwilligen Gruppen entstehen, wenn Gruppenmitglieder entgegengesetzte Interessen haben. Daraus können in virtuellen Gruppen Konflikte zwischen Gruppenausrichtung und einer Minderheit z.B. über verschiedene Ansichten bzgl. der Gruppe, aber auch persönliche Angriffe, Ideendiebstahl, SPAM-Versand und im extremen Fall (sexueller) Belästigung<sup>218</sup> erwachsen, die zu einem Auseinanderdriften bzw. einer Auflösung der Gruppe führen können. Aus diesem Grund werden Konflikte auf Basis der Kooperation kontrolliert und möglichst vermieden, wobei der Dissens über Sympathie und Vertrauen aufgefangen bzw. kompensiert wird. Die Vermeidung hängt in selbstorganisierten Gruppen mit einer Norm der Nicht-Thematisierung der Gruppendynamiken bzw. Bewertungen von anderen Gruppenmitgliedern in einer solchen Gruppe zusammen. Wettbewerb spielt daher kaum eine Rolle. Diese Vermeidung kann beim KDE-Projekt in der teilweise negativen Bewertung der Einführung einer Mailingliste ohne Themenbeschränkung (genannt: KDE-cafe) erkannt werden. Die kooperative, gleichgestellte Selbstreflexion oder Bewertung der selbstorganisierten Gruppe ist wegen möglicher Konflikte tabuisiert, so auch im KDE-Projekt<sup>219</sup> (vgl. Elliott, Scacchi 2003: 10f.; Levine, Moreland 1995: 438; Sader 1991: 105f.; 136f.; Neidhardt 1999: 148; Schattenhofer 1992: 108, 131, 1997: 178; Bernart 1999). Da wie oben beschrieben die Vereinsstruktur kaum Einfluss auf die Aktivitäten hat, finden keine formalen, wettbewerblichen Prozeduren wie Wahlen nach dem Mehrheitsprinzip eine Verwendung.

### Kultursystem mit Professionssystem und Wissenschaftssystem

Eine weitere sehr wichtige gesellschaftliche Umwelt für das Open Source-Projekt KDE stellt das Kultursystem mit Professions- und Wissenschaftssystem dar. Im KDE-Projekt arbeiten hoch spezialisierte Softwareentwickler, die einer eher technisch-naturwissenschaftlichen Profession angehören und eine wissenschaftliche Ausbildung genossen haben oder noch studieren. Fast alle Projektmitglieder sind Informatiker oder kommen aus einem informatiknahen, natur-/ingenieurwissenschaftlichen Bereich (s. Kap. 5.2.). Folglich wurden die freiwilligen Mitglieder in einer Ingenieurs-, Naturwissenschafts- oder Informatiker- (Softwareentwickler-)Profession bzw. Disziplin sozialisiert. Zu-

---

<sup>217</sup> Die starke Aufgabenorientierung des KDE-Projekts betont dabei die aufgabenorientierte Führungsrolle.

<sup>218</sup> Die sexuelle Belästigung von Spielercharakteren tritt aber nur selten in spezifischen virtuellen Gruppen auf, die das Ziel des gemeinsamen Spielens z.B. in Multi User Dungeons haben (vgl. Götzenburger, Löger 2000; Döring, Schestag 2000).

<sup>219</sup> Konflikte können zu einer Abtrennung von Teilgruppen oder Auflösung führen, weswegen diese vermieden werden. Zugleich aber stellen sie Chancen dar, zu einer Weiterentwicklung der Gruppe beizutragen.

sätzlich kann die außeruniversitäre Profession mit der Wissenschaft die wichtigste Ressource der (potenziellen) Mitarbeiter „zur Verfügung“ stellen, d.h. sie sind Hauptrekrutierungsfelder für Neumitglieder. Dies zeigt sich in einem großen Anteil an Studenten im Projekt. Damit können die Koordinationsformen in den außeruniversitären Professionen und den technisch-naturwissenschaftlichen Disziplinen die Koordination im KDE-Projekt beeinflussen.

Die Wissenschaft mit ihren Disziplinen und die außeruniversitären Professionen gehören dem Kultursystem an, da Profession und disziplinäre Wissenschaft auch jeweils über ihre Wertvorstellungen definiert werden. Beide sind wie das KDE-Projekt dem freien Fluss der Informationen bzw. des Wissens verpflichtet (vgl. Kelty 2001; Freidson 2001: 100f.; Abbott 1988: 56). Doch haben beide unterschiedliche Orientierungen. Die Professionsmitglieder unterliegen einem berufsständischen Ethos, den sie sich selbst auferlegt haben. Die Profession hat zusätzlich eine Kunden-/Klientenorientierung und einen direkten Handlungsbezug, d.h. sie bezieht sich auf die Wissensanwendung in der Praxis, außerhalb des Wissenschaftssystems, sowie auf eine Kontrolle des Arbeitsprozesses.<sup>220</sup> Die Wissenschaft ist der Wahrheit und der Erweiterung abgesicherten theoretischen Wissens verpflichtet. Dies umfasst Kodifikation, Verfeinerung, Expansion des Wissens über Theoretisieren und Forschung als zentrale Aufgaben (Stichweh 1994: 17ff., 284ff.; Freidson 2001: 36ff., 132, 214ff.; Abbott 1988: 53ff.; Kelty 2001; Merton 1985: 88ff.; Bühl 1974: 94).

Die Werte und Institutionen der außeruniversitären Professionen und wissenschaftlichen Disziplinen wirken auch außerhalb ihres angestammten, formalen Bereichs. Eher informale „Subgemeinschaften des Spezialwissens“ bilden sich wegen einer voranschreitenden horizontalen Spezialisierung „unterhalb“ der übergreifenden, formalen Ebene der Disziplinen und Professionen, gerade in der Informatik oder den Ingenieurwissenschaften, weiter aus. Das KDE-Projekt kann auch als eine solche Subgemeinschaft bezeichnet werden, da es als informales Projekt die Werte und Institutionen der außeruniversitären Professionen und wissenschaftlichen Disziplinen übernimmt. Dabei erfolgt wie in den außeruniversitären Professionen und wissenschaftlichen Disziplinen eine Ausdifferenzierung, Spezialisierung bzw. Modularisierung im KDE-Projekt (vgl. Luhmann 1990: 446ff., 1992: 446ff.; Stichweh 1994: 18f.; Freidson 2001: 144f.; Abbott 1988: 125).<sup>221</sup> Da die Werte und Institutionen aus Profession und Disziplin auch in Professionssubkultur bzw. Epistemic Communities be-

---

<sup>220</sup> Dabei steht die außeruniversitäre Profession außerhalb des Markts, des Staats und der Organisationen, gleicht deren Nachteile aus und dient dem Schutz des Spezialwissens auf dem Arbeitsmarkt.

<sup>221</sup> Diese eher informalen „Subgemeinschaften des Spezialwissens“ werden auch unter dem Begriff Professionssubkultur (Occupational Subcultures) bzw. Communities of Practice mit der Betonung von Profession und Praxis sowie unter Epistemic Communities mit der Betonung von theoretischem Wissen und Wissenschaftlichkeit diskutiert, wobei die Grenze zwischen den beiden Begriffen fließend ist (vgl. Brown, Duguid 2001: 206; Hakanson 2002: 11). Beispiele für Communities sind mit regionalem Bezugspunkt die „Flat Panel Display Industry“ (vgl. Murtha et al. 2001) oder die vernetzten Ingenieure im Silicon Valley (vgl. Saxenian 1996), mit internationalem Bezug Subcommunities in der Wissenschaft (vgl. Knorr-Cetina 2002). Diese Trennung zeigt sich auch in der Diskussion über Open Source-Projekte, die einerseits als spezifische Professionssubkulturen (Occupational Subcultures) bzw. Communities of Practice (vgl. Elliott, Scacchi 2002, 2003; Tuomi 2002; Holtgrewe 2004b) oder Epistemic Communities (vgl. Edwards 2001; Cohendet et al. 2001) verstanden werden. Dabei entstehen durch die Verbreitung der vernetzten IuK-Technik weitere Subcommunities der Communities of Practice und Epistemic Communities, wobei nicht nur Open Source-Projekte, sondern auch virtuelle Professionen-Communities, die die Karriere und Arbeitsmarktbesetzung ihrer Mitglieder fördern, und neue wissenschaftliche Communities zu Spezialthemen über Mailinglisten entstehen (vgl. Hinner 2003: 88ff.; Benner 2002; Kelty 2001). Bestes Beispiel für eine wissenschaftliche Epistemic Community ist die Community, die sich als Mailingliste und Homepage um das Thema Open Source gruppiert hat (s. <http://opensource.mit.edu>).

stand haben, können die beobachteten Koordinationsformen in diesen Subgemeinschaften auch im Open Source-Projekt auftreten.

Wie oben schon dargestellt sind sich ausseruniversitäre Professionen und wissenschaftliche Disziplinen in bestimmten Punkten sehr ähnlich. Dies liegt auch an einer gemeinsamen historischen Basis, d.h. an einer parallelen Entwicklung aus der vormodernen undifferenzierten Wissenschaft an Universitäten (Bühl 1974; Stichweh 1994). Die außeruniversitäre Profession und disziplinäre Wissenschaft sind miteinander über die Universität verbunden bzw. haben diese als Überschneidungsbereich, wobei die Sozialisation an einer Hochschule eine tragende Rolle in Profession und Wissenschaft spielt. Die Wissenschaftler bzw. Forscher als akademische Professionselite<sup>222</sup>, die sich in Disziplinen korrespondierend zu den Professionen entwickelt haben, bilden die prospektiven Professionsmitglieder theoretisch und praktisch aus (Freidson 2001: 92ff., 1994: 177f.; Abbott 1988: 118f., 129ff., 198ff.; Stichweh 1994: 17ff., 284ff.; Luhmann 1990: 449). An den Universitäten werden z.B. die Studenten in das Programmieren sozialisiert und z.T. an Linux- oder anderer Open Source-Software ausgebildet (vgl. Hornecker 1998).

Gemeinsame Werte und Normen werden beispielsweise in den Einführungskursen der relevanten Disziplinen Informatik, Mathematik, Physik und den Ingenieurwissenschaften sozialisiert. Zudem ist den Disziplinen Informatik, Mathematik, Physik und den Ingenieurwissenschaften die starke Sachorientierung gemeinsam, was Knorr-Cetina (2002) „objektzentrierte Sozialität“ nennt und auch beim KDE-Projekt auftritt. Dies spiegelt sich auch in der studentischen Fachkultur der Ingenieurwissenschaften, Physik oder Informatik wider. Obwohl die Studenten unterschiedliche Fächer studieren, haben diese Fächer jedoch eine ähnliche studentische Fachkultur mit ähnlichen Werten und Normen (Multrus 2004: 378f.).<sup>223</sup> Technische Artefakte stehen im Vordergrund, was im Fall des Open Source-Projekts der Quellcode ist. Das Open Source-Projekt ähnelt zusätzlich diesen Disziplinen z.B. in ihrem dominierenden Männlichkeitsbezug und ihrem Selbstverständnis (vgl. Knorr-Cetina 2002; Laudel 2001; Heintz 2000a: 140).

Neben diesen Überschneidungsbereichen unterscheiden sich Profession und Disziplin in einigen Punkten, was sich verstärkend auf die Übernahme von Koordinationsformen in das KDE-Projekt auswirkt.<sup>224</sup> Diese unterschiedlichen Orientierungen von Profession und Wissenschaft können zu unterschiedlichen Ausprägungen oder einer unterschiedlichen Bedeutung der Koordinationsformen im KDE-Projekt führen, weswegen nacheinander abfolgend und getrennt die Koordination in erstens den wissenschaftlichen Disziplinen mit ihrer Unterform der Epistemic Communities und zweitens den außeruniversitären Professionen mit ihrer Unterform der Communities of Practice/Professionssubkultur thematisiert werden.

Wissenschaftliche Disziplin und Profession wirken unterschiedlich stark auf die Institutionalisierung im Open Source-Projekt ein. Die wissenschaftlichen Werte und Normen wirken deutlich stärker als die der Profession im KDE-Projekt (vgl. Kelty 2001;

---

<sup>222</sup> So hat der Begriff Professor den gleichen Wortstamm wie Profession (Stichweh 1994: 284).

<sup>223</sup> Bei einer quantitativen Befragung von Studenten konnte durch das Clusterverfahren eine studentenbezogene fachkulturelle Nähe zwischen Informatik, Maschinenbau, Elektrotechnik, Bauingenieurwesen, Physik etc. festgestellt werden (vgl. Multrus 2004). Zwischen diesen Disziplinen bestehen geringere Unterschiede in der studentischen und wissenschaftlichen Fachkultur als zwischen diesen und den Geistes-/Sozialwissenschaften, Medizin, Biologie/Chemie, Jura (vgl. Multrus 2004; Freidson 2001, 1994; Stichweh 1994: 24, 278ff.; Heintz 2000a: 190).

<sup>224</sup> Da dies zwei getrennten gesellschaftlichen Akteure aus eigenen organisationalen Strukturen bestehen, verstärken diese beiden Akteure Institutionen im KDE-Projekt besonders; jedenfalls stärker, als wenn nur einer der beiden Akteure Institutionen verkeilen würde.

Raymond 1998b; Bezroukov 1999a, b; Taubert 2006). Beispielweise korrespondiert die Publikation von Texten in der Wissenschaft mit der Publikation von Quellcode im Open Source-Projekt KDE. Der Quellcode in Open Source-Projekten wächst als ein Beispiel genauso exponentiell wie das wissenschaftliche Wissen (vgl. Robles 2004; Koch, Schneider 2002; Hinner 2003; Bühl 1974: 129).

Nach der ersten Sozialisation in den Basiskursen in gemeinsame Werte und Normen werden in einem spezialisierenden Fachstudium das wissenschaftliche Arbeiten, die Arbeitsmethoden und die sozialen Strukturen einer wissenschaftlichen Disziplin internalisiert (vgl. Huber 1991; Multrus 2004, Freidson 2001: 92f., 1994: 177f.; Abbott 1988: 56; Stichweh 1994: 284ff.). Die Studenten erstellen dabei Studien-/Abschlussarbeiten innerhalb eines Forschungsprojekts an einem wissenschaftlichen Hochschulinstitut. Damit wird eher die Wissenschaftskultur institutionalisiert. Die Profession unterstützt die Erhaltung der grundlegenden Werte und Normen nach der Sozialisation an der Hochschule durch einen Wissensaustausch, organisiert durch Berufsverbände. Dabei werden zusätzliche Normen sozialisiert.

Auf die Gründe für die größere Institutionalisierungsstärke der Wissenschaft wird im Folgenden eingegangen. Das Wissenschaftssystem mit Universitäten und Forschungslaboratorien hat von Beginn an die Softwareentwicklung und ihre Profession sowie damit auch die Entstehung der sozialen Open Source-Bewegung mit den Open Source-Projekten wie dem KDE-Projekt mitbestimmt.<sup>225</sup> Dabei spielen und spielten die technisch-naturwissenschaftlichen Disziplinen eine besondere Rolle, da in deren Forschungslabors bzw. Großforschungseinrichtungen von Firmen wie Bell Laboratories von AT&T und von amerikanischen Universitäten wie Berkeley oder MIT das Internet und die eher wissenschaftliche Softwareentwicklung vor 1994 entwickelt wurden. An der Berkeley-Universität wurde z.B. das erste freie Unix-Derivat programmiert, das auf proprietärem, aber freiem Quellcode basierte und aus dem auch Linux entstand. Hauptsächlich Physiker, aber auch Elektrotechnik-Ingenieure, arbeiteten an der Hardware in diesen Forschungseinrichtungen und wurden dann zu Softwareentwicklern. Zudem arbeitete Richard Stallman, ein wichtiger Open Source-Protagonist und Gründer der Free Software Foundation, lange Jahre als Systemprogrammierer am MIT. Die selbstverständliche freie Weitergabe von Software und Wissen als Kooperation dieser wissenschaftsdominierten Frühphase wurde in und durch die Open Source-Bewegung/-projekte und in den Gruppen im Internet konserviert. Außerdem unterstützte die wissenschaftliche Entwicklung des Internets die Verbreitung der Open Source-Projekte und -Bewegung (vgl. Kelty 2001; Holtgrewe, Werle 2001: 51ff.; Grassmuck 2002: 179ff.; Raymond 1999b; Abbott 1988: 39, 93, 181).

Dieser starke wissenschaftliche Einfluss zeigt sich auch bei KDE-Projekt. Einige Rechenzentren der Universitäten bzw. der Forschungslaboratorien wie die Universitäten Tübingen oder Kaiserslautern unterstützen die Softwareentwicklung im KDE-Projekt, indem sie ihre Server für die Software- und Homepageentwicklung im KDE-Projekt zur Nutzung zur Verfügung stellt.<sup>226</sup> Zusätzlich wurde ein Großteil der KDE-Projektbeteiligten in der Informatik, ein paar wenige in der Mathematik bzw. Physik und eher wenige in den Ingenieurwissenschaften sozialisiert (s. Kap. 5.2.1.). Da auch Gründer einen großen Einfluss auf freiwillige Gruppen haben (vgl. Schattenhofer 1992: 134ff.) und ein Gründer Informatik, ein anderer Physik studiert hat, dürfte Gedankengut und Wissenschaftskultur aus diesen Disziplinen Eingang ins KDE-Projekt gefunden haben.

---

<sup>225</sup> "This fellowship [which built around an operating system and a programming environment] was rooted in the traditions of academic research and publication and of telecommunications engineering with its established orientation towards interconnectivity" (Holtgrewe, Werle 2001: 52).

<sup>226</sup> s.a. <http://en.wikipedia.org/wiki/KDE>.

Da die Wissenschaft einen größeren Einfluss auf das KDE-Projekt hat als die Profession, wird erst der Wissenschaft und dann die Profession betrachtet.

### Wissenschaft und Koordinationsformen

Die Koordinationsformenkombination der Wissenschaft bzw. Disziplinen wurde während der Ausbildung an der Hochschule sozialisiert und beeinflusst das Open Source-Projekt KDE. Auf der Ebene der Wissenschaft werden die Beziehungen zwischen den Disziplinen durch Koordinationsformen bestimmt, die in ähnlicher Form und Bedeutung auch in den Wissenschaften auftreten und so im Open Source-Projekt KDE erscheinen. Zwischen den Disziplinen dominiert eine „intellektuelle“ *Kooperation* zwischen Forschern gleicher, aber auch verschiedener Disziplinen, um gegenüber den restlichen Gesellschaftssystemen Ressourcen zu erhalten. Eine *Anerkennungshierarchie* zwischen Disziplinen kann außerdem von den harten (Natur-)Wissenschaften gegenüber den weichen (Geistes- und Sozial-)Wissenschaften gesehen werden, wobei die harten eher Ressourcen bekommen als die weichen (Stichweh 1994: 32). *Wettbewerb* zwischen Disziplinen herrscht um Ressourcen, alleinigen Zugriff auf Untersuchungsbereiche und gesellschaftliche Anerkennung, wobei diese schwer zu beobachten sind und eher als Prioritätsrechte und Widerstand gegen neue Ideen auftritt (Bühl 1974: 97ff.; Poulsen 2001).<sup>227</sup>

Bei einer Fokussierung der beeinflussenden Wissenschaft zeigt sich wiederum der größere Einfluss der (Grundlagen-)Wissenschaft gegenüber einer Praxisorientierung, die sich an die Profession anlehnt. Betrachtet man die Disziplinen Informatik, Mathematik, Physik und der Ingenieurwissenschaften näher, dann unterscheiden sie sich abgesehen von den grundlegenden Normen durch ihre gleichzeitige Orientierung sowohl für angewandte, praxisnahe Lösungen als auch für Grundlagenforschung/-theorie. Dominiert in der Ingenieurwissenschaften aber die Anwendung und praxisbezogene Gestaltung, so dominiert in Mathematik und Physik die grundlagenwissenschaftliche Kultur.<sup>228</sup> Die Informatik, deren Wissenschaftskultur durch die vielen Informatiker im KDE-Projekt dieses Projekt mitbestimmt, kann weder eindeutig der grundlagenwissenschaftlichen noch der praxisbezogenen Orientierung zugeordnet werden. Die Vertreter der Informatik zählen sich entweder eher zur Formalwissenschaft (Mathematik) als Grundlagenwissenschaft oder eher zu den Ingenieurwissenschaften mit ihrer Praxisorientierung (vgl. Büttemeyer 1995: 91ff.; Bittner 2003).

Doch hat die Wissenschaftskultur der Grundlagenwissenschaften Mathematik, Physik einen größeren Einfluss auf das Open Source-Projekt als die der angewandten Ingenieurwissenschaften mit ihrem Gestaltungsbezug. Zwar durchlaufen die Studenten an den Hochschulen in der Informatikdisziplin größtenteils eine praxisbezogene Ausbildung mit formalen Konstruktions- und Entwurfsmethodiken (vgl. in Kap. 2.3. die formalen Vorge-

---

<sup>227</sup> Neuerdings geraten auch die Hochschulen unter Wettbewerbsdruck durch Politik und Wirtschaftsverbände (vgl. Krücken 2004), der hier aber als neue Entwicklung nicht betrachtet wird.

<sup>228</sup> Physik und Mathematik sind reine Grundlagenwissenschaften, die auf der Suche nach Wahrheit, d.h. theoretischer Erkenntnis sind. Dabei spielt der wissenschaftliche Versuchs- und Irrtums-Prozess in der Forschung wie im KDE-Projekt eine Rolle. Das Erkennen dessen, was ist, bzw. das wissenschaftliche Vorgehen haben Physik und Mathematik mit der Ingenieurwissenschaft gemeinsam. Die Ingenieurwissenschaften als Technikwissenschaften konzentrieren sich nicht auf das Erkennen, sondern auf das Gestalten oder neue Realitäten kreieren, d.h. „sie arbeiten technische Möglichkeiten aus und empfehlen deren Überführung in die technisch-gesellschaftliche Praxis“ (Ropohl 2004: 28f.). Sie machen dazu eine praxisorientierte Modellbildung, um die Welt verändern zu können. Ihr Endziel ist dabei der Erfolg und die praktische Umsetzungs- und Funktionsfähigkeit. Weiterhin ist die universitäre Ausbildung auf das Entwickeln und Produzieren in einem Unternehmen ausgerichtet (vgl. Ropohl 2004: 28f.; Banse et al. 2006).

hensmodelle für die Softwareentwicklung in Unternehmen). Doch ist die Softwareentwicklung nur ein Fach unter vielen. Ein Grossteil der Fächer dreht sich eher um einen Versuchs-und-Irrtums-Prozess, wie er in den Grundlagenwissenschaften auftritt (vgl. Broy, Schmidt 1999). So bestimmen besonders Werte der Grundlagenwissenschaften (Mertons wissenschaftlicher Ethos) den Arbeitsprozess in Open Source-Projekten (vgl. Taubert 2006: 105; s.a. Glock 1997).

Damit der Einfluss der Grundlagenwissenschaften auf das KDE-Projekt dargestellt werden kann, wird die Wissenschaftskultur und institutionelle Struktur in Großforschungseinrichtungen der Physik- bzw. Mathematikdisziplin in Ermangelung umfangreicher Untersuchungen bzgl. des wissenschaftlichen Vorgehens in der Informatik fokussiert. Außerdem bestehen die Großforschungseinrichtungen neben den Universitätsstrukturen und sind daher wenig durch diese beeinflusst. Sie sind eigenständige Organisationen, in denen wegen der disziplinären Ausrichtung keine Vermischung mit anderen Disziplinen besteht und so die grundlegenden Strukturen dieser Grundlagenwissenschaften (Physik- oder Mathematikkultur) unverstellt, in Reinform wiedergegeben werden. Auch da die Wissenschaftskultur der Ingenieurwissenschaften große Überschneidungen mit den anderen Kulturen hat, wird nicht gesondert auf diese eingegangen. Wenn aber der Gestaltungsaspekt eine Rolle spielt, wird dieser gesondert aufgeführt.

Auf der Basis der allgemeinen Wissenschaftskultur und der Erkenntnisse der Großforschungseinrichtungen der Physik und Mathematik können sozialisierte Ausprägungen und die Bedeutung der Koordinationsformen festgestellt werden, die auch im Open Source-Projekt KDE angetroffen werden können:

- *Kooperation* mit einem Grundkonsens der Wissensweitergabe bestimmt die Wissenschaft, wobei alle Forschenden als gleich angenommen werden. Dies gilt gerade für die amerikanische Wissenschaft, aus der das Internet und Open Source-Projekte entstanden (s.o.). Die Forscher können ihre Forschungsthemen frei wählen und ihre Meinung frei äußern. Wie im KDE-Projekt findet eine herrschaftsfreie Kommunikation ohne eine Entscheidungscentralisierung statt, d.h. ein offener, anonymer Zugriff auf Wissen und eine Toleranz gegenüber Meinungsverschiedenheiten/Konflikten tritt neben einem fehlenden, terminierten Entscheidungsdruck auf (Luhmann 1990: 319, 365; Bühl 1974: 95; Kelty 2001; Merton 1985; Taubert 2006).

Eine reziproke Kooperation zeigt sich generell im Wissenschaftssystem. Forscher oder Forschergruppen der gleichen Disziplin kooperieren vor allem innerhalb und zwischen Forschungsinstitutionen, wobei in Deutschland z.T. über institutionelle Grenzen in Sonderforschungsbereichen interdisziplinär geforscht wird.<sup>229</sup> Diese Kooperationen bestehen temporär bzgl. gemeinsamer, sachorientierter, wissenschaftlicher Probleme, Methoden oder Untersuchungsobjekten an allen Punkten eines Forschungsprozesses mit gleichberechtigten Partnern, wenn diese wie im KDE-Projekt nicht alleine lösbar sind (vgl. Laudel 2001; Laudel, Gläser 1998; Heintz 2000a).

Weiterhin gibt es Lehrer-Schüler-/Professor-Doktoranden-Verhältnisse mit reziproker Kooperation, bei denen der Lehrer als Experte den Schüler berät und der Schüler dem Lehrer bei der Forschung hilft (vgl. Wildavsky 1986; Poulsen 2001; Laudel 2001). Koautorchaft, die aus einer sachorientierten Kooperation entsprang, aber auch Zitationskartelle sind ein weiteres Merkmal für reziproke Kooperation wie im KDE-Projekt. Besonders in der Formalwissenschaft arbeiten die einzelnen Mathematiker eher allein und kooperieren spontan in temporären, informellen Arbeitsgruppen, Gesprächen und Seminaren (vgl. Hin-

---

<sup>229</sup> In diesen Sonderforschungsbereichen bilden sich ähnliche kooperative Strukturen wie in Universitäten, Forschungsinstituten oder wissenschaftlichen Disziplinen aus (vgl. Laudel, Gläser 1998).

ner 2003: 159; Knorr-Cetina 2002: 236, 256ff.; Laudel 2001; Heintz 2000a: 190ff.). Die Publikation von Texten, Aufsätzen, Büchern wird dabei entweder im Sinne der Austauschtheorie (Publikation gegen Anerkennung) oder als Geschenkmodell nach Mauss verstanden, bei der nach dem Geschenk die Anerkennung aus „Gnade“ gewährt wird (vgl. Hagstrom 1982).<sup>230</sup>

Es tritt nicht nur eine reziproke Kooperation, sondern auch eine bedeutende gruppenorientierte Kooperation wie im KDE-Projekt auf (vgl. Bühl 1974: 72, 74ff., 79ff.). Konferenzen, informale Workshops, Mailinglisten und Projektbesprechungen bzw. Diskurszirkel in Forschungsgruppen bzw. an Hochschulen sind Beispiele für diese gruppenorientierte Kooperation in der Wissenschaft, bei der Wissen zwischen Forschergruppen mit unterschiedlicher Distanz vom Forschungsgegenstand, aber der gleichen Disziplin weitergegeben wird. Die Wissensweitergabe erfolgt ähnlich im KDE-Projekt über das Chatsystem, Mailinglisten und die alljährliche Konferenz. Vertrauen, Sympathie und die Aussicht auf gegenseitigen Wissensaustausch spielen in Wissenschaft und KDE-Projekt eine große Rolle (vgl. Poulsen 2001; Heintz 2000a: 140, 192f.; Knorr-Cetina 2002: 236, 256ff.).

Eine bedeutende, gruppenorientierte Kooperation mit spezifischen Vertrauensbeziehungen (Klatschzirkeln) konnte Knorr-Cetina (2002: 227; 234f., 282ff.)<sup>231</sup> bei Physikern und Heintz (2000a: 182; 188ff.) bei Mathematikern feststellen. Sie spielt auch beim KDE-Projekt eine große Rolle. Die gruppenorientierte Kooperation ist in der Mathematik besonders stark ausgeprägt, da sich alle Wissenschaftler, obwohl eine starke Spezialisierung besteht, als Mathematiker verstehen. Beide Wissenschaftskulturen stehen dabei im Gegensatz zur Kultur der Molekularbiologen mit einer vorherrschenden, reziproken Kooperation (Knorr-Cetina 2002: 302ff., 323ff.). Auch wird bei Entscheidungen in der Physikwissenschaft und der Mathematik nach der „vernünftigsten“ Lösung gesucht (Knorr-Cetina 2002: 256, 275; Heintz 2000a: 153, 177ff.).<sup>232</sup> Dies tritt auch wie oben beschrieben bei den KDE-Entwicklern bei der Suche nach der technisch praktikabelsten Lösung auf (vgl. Ettrich 2004; für andere Open Source-Projekte: Yamauchi et al. 2000: 7). Weiterhin besteht eine kooperative Teamarbeit in Form von formalen, aber auch informalen Gruppen mit einer Zentralisierung von Autorität und Verantwortung auf Projektleiter auch an Universitäten (vgl. Bühl 1974: 193ff.; Poulsen 2001). Während sich Kooperation in der Forschungseinrichtung der Physik eher in und zwischen formalen Gruppen zeigt, ist die Forschungseinrichtung der Mathematik generell auf Ad hoc-Kooperationen zwischen informalen Gruppen ausgerichtet, wobei beide Kooperationsarten im KDE-Projekt auftreten können (Knorr-Cetina 2002: 227ff., 308ff., 323ff.; Heintz 2000a: 140f., 192f.).

- Zusätzlich herrscht eine *Statushierarchie* durch Reputation, die über neue Erkenntnisse in Publikationen erworben wird. Es bildet sich ein Kern von bekannten Wissenschaftlern heraus, was auch im Open Source-Projekt KDE geschieht. Über Publikationen erhalten die Wissenschaftler Reputation, über die sie wiederum spezielle Posten erhalten. Diese speziellen Posten dienen wie im KDE-Projekt der Kontrolle von Kollegen, was unter dem Begriff Selbstkontrolle der Wissenschaft bekannt ist. Dabei kontrollieren z.T. formale Posten als „Torwächter der Wissenschaft“ wie Zeitschriftenherausgeber, wissenschaftliche

---

<sup>230</sup> Man bemerke die ähnliche Argumentationsweise wie im wissenschaftlichen Diskurs um die Einordnung der Zusammenarbeit in Open Source-Projekten (s. Kap. 5.4.1.).

<sup>231</sup> Diese Kooperation der Physiker bezeichnet Knorr-Cetina (2002) als kommunitaristisch, da eine Gemeinschaft über eine Sache entsteht und vom Problem bzw. Experiment her gedacht wird. Jeder hilft durch seinen Input.

<sup>232</sup> In der Physik zeigt sich diese Suche vor allem am Anfang eines Experiments in der Auswahl der geeigneten, und optimal erscheinenden Technik und nicht in der Zusammenstellung von Wissenschaftlern bzw. der Auswahl einer bestehenden wissenschaftlichen Gruppe nach ihrer Reputation oder anderer personenbezogener Kriterien (Knorr-Cetina 2002: 256, 275).

Gutachter, Doktorväter und Rezensenten das Publizieren von Texten wie die Beantragung von Forschungsprojekten (Bühl 1974: 87f.; Merton 1985: 172ff.; Heintz 2000a: 159, 180). Dabei ist die Kontrolle durch Kooperation im Peer-review-Verfahren als gegenseitige Kontrolle zwischen Autor und Kontrolleur überformt, da der Kontrolleur auch wieder als Autor auftritt (vgl. Hinner 2003: 129f.; Bühl 1974; Merton 1985: 202f.; Heintz 2000a: 181f.). Dieses Verhältnis zwischen Gutachter und Autor wird zudem durch die neue, freie Veröffentlichung im Internet und die gleichgestellte Bewertung durch die Leserschaft egalisiert (Hinner 2003: 132ff.).

Dazu gesellen sich auch formale Posten, wie sie an Universitäten oder anderen Forschungseinrichtungen üblich sind. Normalerweise besteht eine Hierarchie vom Professor und/oder Laborleiter über wissenschaftliche Mitarbeiter/Doktoranden, Verwaltungsmitarbeiter zu den Studenten bei Examina bzw. Hausarbeiten und vom Gutachter oder Zeitschriftenherausgeber zum Autor (vgl. Stichweh 1994; Bühl 1974). Beim KDE-Projekt übernehmen ähnliche Posten in Form von Projektleitern oder dem Releasekoordinator diese Torwächter-Funktion. Diese Art der Hierarchie/Macht von Posten rangiert wie im KDE-Projekt zwischen reiner Kontrolle und direkter Mitarbeit wie beim hierarchischen Verhältnis zwischen Professor und Doktorand an Universitäten, wobei der Professor oftmals nur überwacht und nicht mitforscht, im Gegensatz zu nicht-universitären Forschungsinstituten/-laboren, bei denen der Forschungsgruppenleiter oftmals an der Forschung beteiligt ist (vgl. Laudel 2001: 767). Da aber im Allgemeinen die Studenten nicht an der Hochschule bleiben und nicht stärker in die hierarchische Struktur einer Universität eingebunden werden, gehen die sozialisierten, formalen, universitären Strukturen wie diese Hierarchie/Macht ausgehend vom Professor in das KDE-Projekt zwar auch, aber eher schwach ein. Eine eher geringere Hierarchie/Macht durch Expertenwissen und ständige Anwesenheit der Posteninhaber an Besprechungen tritt in den Wissenschaftsstrukturen der Physiker und Mathematiker auf (Heintz 2000a: 140f., 192f.; Knorr-Cetina 2002: 227ff.), was sich dadurch eher im KDE-Projekt niederschlägt.<sup>233</sup>

- In der Wissenschaft existiert außerdem ein wenig bedeutender *Wettbewerb* um zu bewilligende Forschungsprojekte/-gelder, talentierte Studenten, Nutzung von teuren Apparaten/Großforschungseinrichtungen, Universitätsstellen, Mitgliedschaft in wissenschaftlichen Vereinigungen, um Ehrungen/Preise und Herausgeberschaften (vgl. Edge 1990). Dies gilt nicht so umfassend für das Open Source-Projekt KDE, in dem nur wenige Posten existieren oder Ehrungen vorgenommen werden. Außerdem herrscht ein Wettbewerb um die schnelle Publikation oder Urheberschaft von neuem Wissen und um Zitationen, was auch für das KDE-Projekt im begrenztem Maße gilt (Merton 1985: 97; Luhmann 1990: 246ff.; 576ff.; ; Heintz 2000a: 160ff.).<sup>234</sup> Die Zitationen haben in Open Source-Projekten ihre Entsprechung in der Nennung als Autor in der GPL-Softwarelizenz (s. Kap. 5; Kelty 2001).

Das Medium des Wettbewerbs ist in der Wissenschaft die Reputation, genauso wie bei Open Source-Projekten (Kelty, 2001; Raymond 1998b; Bezroukov 1999a, b). Die Reputation wird bei den Physikern ähnlich wie beim KDE-Projekt durch kleine Workshops in Form von informalen Treffen, Gerüchte/Klatschzirkel, Beobachtung auf Konferenzen, Zugehörigkeit zu reputationsträchtigen Experimenten bzw. Subprojekten vermittelt (vgl. Knorr-

---

<sup>233</sup> Als Gegenbeispiel sei hier die höhere Hierarchie/Macht bei Molekularbiologen genannt, da über den Posten des Laborleiters alle Außenkontakte laufen, die Labormitarbeiter von seinen Informationen sowie seiner Außendarstellung auf Konferenzen abhängen und Publikationen sowie Reputation mit dem Laborleiter identifiziert werden (Knorr-Cetina 2002: 308ff., 323ff.).

<sup>234</sup> Zitation und Zitationsindices bestehen zwar in der Wissenschaft bzw. können durch neue Auszählwerkzeuge des Quellcodes in Open Source-Projekten etabliert werden (Kelty, 2001), spielen aber nach meiner empirischen Beobachtung im KDE-Projekt kaum eine Rolle.

Cetina 2002: 266f.; Heintz 2000a). Dies erfolgt beim KDE-Projekt ähnlich über Chats oder auf persönlichen Treffen bzw. Konferenzen, z.T. auch über Mailinglisten. Mit Reputation tritt auch in beiden Fällen ein Matthäus-Effekt ein, d.h. wer schon Reputation hat, bekommt noch mehr und hat damit einen immer besseren Zugang in der Wissenschaft zu talentierten Studenten/Doktoranden, finanziellen Mitteln, Posten, Publikationsmöglichkeiten, was im Falle des KDE-Projekts für Posten und vermehrte Reputation gilt (Merton 1985: 94, 97, 147; Luhmann 1990: 251; Poulsen 2001; Brand, Holtgrewe 2007). Ein Wettbewerb zwischen den Wissenschaftlern findet wie beim KDE-Projekt aber eher latent statt, d.h. es wird nicht darüber gesprochen, da eine Norm der Bescheidenheit herrscht. Die Diskussion/Wissenskommunikation der Wahrheitssuche wird zwar über den Dissens angetrieben, dieser führt aber auf lange Sicht zu einem Konsens (vgl. Edge 1990; Luhmann 1990: 251; Merton 1985: 276ff.). Die Reputation führt durch die neuen IuK-Techniken zu einer neuen Wissensflut, „Überproduktion“ und Modularisierung von Publikationen und einer weiteren Differenzierung der Wissenschaft, was einer Beurteilung der Reputation von einzelnen Wissenschaftlern entgegensteht (vgl. Hinner 2003: 159ff.).

### Profession und Koordinationsformen

Die Koordinationsformen werden normalerweise innerhalb von spezifischen Professionen sozialisiert und dann in das KDE-Projekt getragen. Sie treten aber nicht nur professionsintern, sondern auch in den Beziehungen zwischen einzelnen außeruniversitären Professionen auf. Die Beziehung zwischen den außeruniversitären Professionen, bestehend aus spezifischen Koordinationsformen, spiegelt dabei das Institutionengeflecht der Koordinationsformen in einer Profession wider und bestimmt somit Ausprägungen und Bedeutung der Koordinationsformen im KDE-Projekt.

Die einzelnen außeruniversitären Professionen unterscheiden sich durch ihr formales Spezialwissen, wobei eine funktional-horizontale und vertikale Arbeitsteilung die Beziehung zwischen diesen bestimmt. Eine horizontale Arbeitsteilung mit *Kooperation* zeigt sich durch eine Aufgabenaufteilung verschiedener gleichgestellter Berufsgruppen, wie Maler, Installateure, Dachdecker, etc. beim Hausbau, gleichen politischen Interessen gegen finanziellen Kürzungen im Gesundheitswesen oder wenn eine Profession keine Antworten zu bestimmten Problemen geben kann. Eine vertikale Arbeitsteilung mit *Hierarchie/Macht* zeigt sich in ihrer funktionalen Wichtigkeit für die Öffentlichkeit, ihrem Expertenwissen und ihrer Überzeugungskraft gegenüber dem Gesetzgeber. Beispielsweise dominiert eine Profession wegen ihres Spezialwissens andere, z.T. Nicht-Professionen (Wissenshierarchie) z.B. im Krankenhaus die Ärzte die PflegerInnen, Experten gegenüber Nicht-Experten, Professionen gegenüber Technikern/Handwerkern. Weiterhin besteht *Wettbewerb* zwischen den außeruniversitären Professionen um die Anerkennung und Ressourcen des Staates bzw. der öffentlichen Meinung/Klienten. Doch hat Wettbewerb eine geringe Bedeutung, da dieser zwischen einer Profession und den Arbeitnehmern bzw. anderen Professionsmitgliedern durch soziale Schließung vermieden wird (Freidson 2001: 52ff., 201f.; 1994: 65ff., 174f.; Abbott 1988: 47ff., 56, 59ff., 82). Beim Open Source-Projekt KDE kann ähnlich zwischen den Professionen eine soziale Grenze mit einer vertikalen Beziehung zwischen den zentralen Softwareentwicklern, den peripheren Dokumentaren/Übersetzern und Grafik/Sound/Design-Entwicklern sowie zwischen Experten und Nicht-Experten wie Endnutzern wiedergefunden werden. Nur wenige Softwareentwickler arbeiten zugleich auch an anderen Tätigkeiten (s. Kap. 5).

Im Weiteren wird die professionsinterne Koordination fokussiert, die am ehesten durch das Studium sozialisiert wird und dadurch besonders die Koordinationsformen im KDE-Projekt verkehrt.

- Innerhalb von Professionen und Communities of Practice bzw. Occupational Subcultures besteht eine spezifische Kombination von Koordinationsformen, in denen kooperative Elemente dominieren (vgl. Hakanson 2002; Wenger 1998; Freidson 1994, 2001; Abbott 1988). Basis der (gruppenorientierten) *Kooperation* ist eine starke Solidaritätsnorm, gepaart mit einem gemeinsamen Berufsethos als Grundkonsens, wobei die Vermeidung von professionsexterner und -interner Kritik im Rahmen eines Korpsgeistes (Leben-und-Leben-lassen-Etikette)<sup>235</sup> und die Norm der formalen Gleichheit eine Rolle spielt. Unterstützend wirkt die Entwicklung einer Professionsidentität bzw. Gemeinschaft auch durch ein gemeinsames Training in Kohorten, getrennt von anderen Disziplinen, für eine bestimmte längere Zeit mit prägender gemeinschaftlicher Sozialisation in gleichen Basiskursen. Weiterhin wird in den Professionen Expertenwissen generiert und kooperativ vermittelt. Über Kommunikationskanäle in Form von Zeitschriften von Professionsverbänden, aber auch informale Stammtische, kleine Runden oder neuerdings Mailinglisten werden Probleme und Anwendungen des Wissens, über nachuniversitäre praxisorientierte Trainings-/Seminarprogramme und Informationen über neueste Entwicklungen ausgetauscht.<sup>236</sup> Diese Gruppen bildeten sich wegen des Auftretens gleicher Aufgaben oder Probleme, aber auch wegen der Norm der freiwilligen Weitergabe von Wissen, die mit intrinsischer Motivation und einer Autonomie im Hinblick auf die Gestaltung der Arbeit einhergeht (Freidson 2001: 100f., 129ff., 1994: 129ff., 140ff.; Abbott 1988: 150ff.). Manchmal wird die kooperative Hilfe eines Experten zu bestimmten Spezialproblemen der Praxis hinzugezogen (Abbott 1988: 40f.).<sup>237</sup> Dies ähnelt stark dem Vorgehen im KDE-Projekt.

- Innerhalb einer Profession tritt aber auch eine Spezialisierung des Wissens auf, die sich in Expertenmacht für diesen Spezialbereich ausdrückt. Zusätzlich besteht eine Meritokratie wie im KDE-Projekt, die sich in einer *Reputationshierarchie* darstellt (Freidson 1994: 104). Vor allem Wissenschaftler, also „Professionsprofessionelle“, haben durch ihr Spezialwissen einen hohen Status (vgl. Abbott 1988: 118). Ein weiterer Machteinfluss geht von Professionsverbänden aus, wobei Professionsmitglieder ihre zukünftigen Mitglieder selbst auswählen und sich auch selbst kontrollieren (Peer Review).

Auf formaler Ebene bildeten sich hauptsächlich bei großen Professionen wie bei den Juristen und Medizinern Professions-Assoziationen mit einem Disziplinarausschuss<sup>238</sup>, während eine informale, professionelle Selbstregulation durch den Druck von Meinung und Tradition weiterbesteht. Da aber die Informatik und die Ingenieurwissenschaften fragmentiert sind, gibt es zwar formale Berufsverbände, aber ohne formale Sanktionsmacht wie bei Juristen oder Medizinern, sondern eine Sanktionierung erfolgt eher informal wie im KDE-Projekt (Freidson 2001: 101, 167ff.). Auch wenn eine mächtige Professionsvertretung be-

---

<sup>235</sup> Eigentlich: „Eine Krähe hackt der anderen kein Auge aus.“

<sup>236</sup> Die Profession umfasst die Sozialisation bzw. Imitation von Normen nach einer Sozialisation an einer Hochschule. Daher wird hier von einer außeruniversitären Profession gesprochen.

<sup>237</sup> Nach Abbott (1988: 40ff.) zerfällt die Arbeit eines Professionsmitglieds bei seiner Konsultation in die Abfolge Diagnose, Behandlung und Schlussfolgerung (Weiterentwicklung des Wissens/Theorieentwicklung), wobei nicht nur eine Beziehung zwischen Klient und Professionsmitglied besteht, sondern auch zwischen dem konsultierten Professionsmitglied und anderen Professionsmitgliedern besonders bei Diagnose und Schlussfolgerung.

<sup>238</sup> Eine mächtige Professionsvertretung wie bei Juristen oder Medizinern erhält diese Vertretung Macht vom Staat für die Sanktionierung. Die Verwaltungsleute einer solchen Professionsvertretung haben einerseits durch ihre gesonderte Vollzeit-Stellung einen größeren Einfluss als die normalen Professionsmitglieder, andererseits haben sie Zugriff auf Sanktionsmechanismen.

stehen würde, würden Sanktionen selten angewendet, vor allem die extreme Sanktion des Ausschlusses von Mitgliedschaft und Privilegien der Profession erfolgt sehr selten, da eine Norm des Leben-und-Leben-lassen bzw. der allgemeinen Gleichheit existiert (Freidson 2001: 168ff., 1994: 129ff., 142, 174f.; Trice, Beyer 1993: 205; Abbott 1988: 125, 150ff.). Ähnliches gilt für die Wissenschaft, die innerhalb ihres Subsystems den Umgang mit Wissen nur durch Ausschluss sanktionieren kann, nicht jedoch außerhalb ihres Systems (vgl. Hinner 2003).

- Innerhalb der außeruniversitären Profession tritt zwar ein *Wettbewerb* um zahlende Kunden wie Privatpersonen oder Organisationen auf, der aber durch eine Kontrolle des Zugangs zur Profession begrenzt wird. Mit einer Meritokratie bzgl. Arbeitsqualität, Beiträgen zur Professionspraxis und Verbesserung der Disziplin geht der verdeckte Wettbewerb um Reputation, symbolische Belohnungen bzw. kollegialen Respekt wie im KDE-Projekt einher, da offener Wettbewerb nicht gern gesehen wird. Wettbewerbsarten wie Ausschreibungen oder Werbung sind für die Ingenieur- oder Informatikprofession aber nicht gesetzlich verboten, wie bei Ärzten oder Juristen (vgl. Freidson 2001: 203, 1994: 66ff.; Parsons 1949).

### Politiksystem und soziale Open Source-Bewegung

Das Politiksystem besteht u.a. aus Staat/staatlicher Verwaltung, Parteien, juristischem System und sozialen Bewegungen, wobei diese Teilsysteme als Gemeinsamkeit auf einen Interessensausgleich ausgerichtet sind (s.o.). In diesem Abschnitt steht aber die Beziehung der sozialen Open Source-Bewegung zum Open Source-Projekt KDE im Mittelpunkt.

Die soziale Bewegung gehört zum Politiksystem, da diese das öffentliche Gut des frei zugänglichen Quellcodes zum Wohle der Gesamtgesellschaft gegen die partikularen Interessen der Wirtschaftsunternehmen zu schützen versucht. Das KDE-Projekt gehört zur sozialen Open Source-Bewegung, die aus heterogenen Elementen wie normativen Bewegungsververtretungen, Infrastrukturprojekten und anderen Open Source-Projekten besteht (s.a. Kap. 5.1). Die soziale Open Source-Bewegung stellt also offensichtlich eine gesellschaftliche Umwelt für das KDE-Projekt dar, aus dem das Open Source-Projekt potenzielle Mitarbeiter als Personal erhält.

*Sozialisiert* in die Werte der sozialen Open Source-Bewegung werden wahrscheinlich überwiegend Endnutzer der Software, die somit mit der Open Source-Bewegung und ihren Ideen vertraut werden. Die potenziellen Mitglieder verfolgen u.a. die Diskussionen in der Open Source-Bewegung oder kennen daher die Werte, Normen bzw. die Softwarelizenz GPL der Open Source-Bewegung (s. Kap. 5; vgl. Della Porta, Diani 1999: 112ff.). Weitere neue Mitglieder werden manchmal durch die Arbeit in anderen Open Source-Projekten sozialisiert. Die Gründer des KDE-Projekts, aber auch weitere Mitarbeiter, kamen z.B. aus dem Open Source-Projekt Lyx.<sup>239</sup> Aus anderen Open Source-Projekten wechselten aber später eher nur vereinzelt Personen ins KDE-Projekt.

Die soziale Open Source-Bewegung eint die Forderung nach einem freien Fluss des Wissens und protestiert gegen eine voranschreitende Proprietarisierung von Wissen durch wirtschaftliche Akteure<sup>240</sup>, wobei dieses Verständnis aus der Wissenschaft und der außeruniversitären Profession übernommen wurde (s. Kap. 5; Zimmermann 2004; Scacchi, Elli-

---

<sup>239</sup> Im Lyx-Projekt wird eine einfache grafische Oberfläche für das komplizierte Desktop Publishing-Programm Latex entwickelt, das über eine Befehlszeile programmiert wird und auch unter der GPL steht (s. <http://www.lyx.org/about/license.php>).

<sup>240</sup> Dies zeigen sie durch eine Verlinkung zur Homepage der Protestorganisation gegen eine Proprietarisierung von Wissen in der Europäischen Union (s. [www.kde.org](http://www.kde.org)).

ott 2004).<sup>241</sup> Diese Gegnerschaft gegen eine Proprietarisierung von Wissen wurde durch die Imitation des Konstrukts Softwarelizenz und ihrer trickreichen Veränderung juristisch festgeschrieben (vgl. Jaeger, Metzger 2002). Die Free Software Foundation (FSF) entwickelte die Softwarelizenz GPL (vgl. Kap. 5.1.3.), die mittlerweile in Deutschland von Gerichten, also dem juristischen System, anerkannt wurde (c't 2004a, b). Die GPL-Lizenz wurde bei einigen Open Source-Projekten als letztes Mittel vor Gericht gegenüber Quellcodeverwendung von Großunternehmen eingesetzt. Die FSF hat aber wiederum durch die Lizenzierung unter der GPL die Möglichkeit, über Gerichte missbräuchliche Verwendungen der Softwarelizenz im KDE-Projekt oder anderen Projekten zu *sanktionieren*, d.h. die lizenzkonforme Nutzung der Software wird von der Foundation im KDE-Projekt kontrolliert. Die Einhaltung der Lizenz wird aber hauptsächlich durch schlechte Publicity in den Medien als weitere Sanktion erreicht, vor allem auf den journalistischen Plattformen der Infrastrukturprojekte der Open Source-Bewegung.

Das KDE-Projekt *imitierte* auch andere Open Source-Projekte und Organisationen der sozialen Bewegung, um bei diesen und auf der politischen Entscheidungsebene gehört und akzeptiert zu werden, vor allem wenn es um politische Inhalte, wie die Änderungen des (europäischen) Urhebergesetzes geht (vgl. Kap. 5.1.4.2.; Claessens 1983: 494; O'Mahony 2003a, b).

Generell sind die KDE-Mitglieder also gegen eine Proprietarisierung von Software. Obwohl das KDE-Projekt eher eine pragmatische Wahl der Mittel zur Durchsetzung ihrer Gegnerschaft propagiert (s. Kap. 5.5; Brucherseifer 2004: 72), zeigt das KDE-Projekt seine Zugehörigkeit zur Bewegung, indem der Quellcode unter die wichtigste Softwarelizenz (GPL) gestellt wurde. Das KDE-Projekt übernahm bzw. imitierte die Softwarelizenz GPL dabei von der FSF, um die Anerkennung und Glaubwürdigkeit des KDE-Projekts gegenüber der Bezugsgruppe aus anderen Open Source-Projekten (s.u.) und gegenüber der sozialen Bewegung zu erreichen und erhalten. Zusätzlich übte die FSF mit weiteren Anhängern informalen Druck über Diskussionen, Aberkennung des Status einer freien Software und Schmähungen auf das KDE-Projekt aus. Dieser Druck erfolgte bei der Lizenzierung der Basisbibliothek des KDE-Projekts, die von der Firma Trolltech zur Verfügung gestellt wurde. Schritt für Schritt wurde diese Basisbibliothek zuerst unter einer eigenen Lizenz (QPL)<sup>242</sup>, schließlich unter der GPL lizenziert.

Über die Anerkennung des KDE-Projekts durch die gemeinsame Lizenz kann erstens der Zustrom von intrinsisch motivierten, potenziellen Mitarbeitern als Personal unterstützt werden. Zweitens wird über die Lizenz die Open Source-Software des KDE-Projekts rechtlich abgesichert. Damit können die eigenen Interessen des freien Zugangs zur Software und der Bewahrung des öffentlichen Guts-Charakters glaubwürdig gegen die Proprietarisierung durch Unternehmen durchgesetzt werden. Zudem bekommt das KDE-Projekt durch die Übernahme einer Lizenz Legitimität vom juristischen System falls eine Sanktionierung ansteht. Die Open Source-Bewegung unterstützt eine informale, aber auch formale, juristische Sanktionierung von Lizenzübertretern (s.o.).

Wie oben beschrieben werden einige potenzielle Mitarbeiter in die Normen bzw. Institutionen, darunter Koordinationsformen, der sozialen Bewegung sozialisiert, die sie in das KDE-Projekt mitbringen. Das KDE-Projekt imitiert aber auch Protestaktionen und Beein-

---

<sup>241</sup> Damit unterscheidet sich diese Bewegung z.B. von der Ökologiebewegung, die eher skeptisch gegenüber den Wissenschaften ist (Della Porta, Diani 1999; Brand 1998; Rucht et al. 1997).

<sup>242</sup> s. <http://www.trolltech.com/developer/licensing/qpl.html>

flussungen auf die politische Entscheidungsebene. Außerdem kann die Bewegung das KDE-Projekt über das juristische System und über Unterstützungsverlust sanktionieren.

Dabei schließt die Softwarelizenz GPL bestimmte Sanktionsarten wie den Entzug von Software zwischen den Softwareentwicklern aus und unterstützt dabei u.a. dadurch die eher *kooperative* Struktur der Softwareentwicklung im KDE-Projekt (s. Kap. 5.1.). Die Vorschrift, dass Quellcode wieder veröffentlicht werden muss, legt damit eine Kooperation zwischen vorherigem und nachfolgendem Programmierer fest. In einer immerwährenden Kette kann der nachfolgende Programmierer auf geschriebenen Quellcode als Gabe zurückgreifen, keiner kann ausgeschlossen werden. *Wettbewerb* zwischen programmiertem Quellcode tritt nur in dieser Form auf und wird nicht ausgeschlossen, da jeder den Quellcode verändern darf. Wettbewerb mit einer proprietären Schließung des Quellcodes findet nicht statt. Wettbewerb zwischen programmiertem Quellcode führt aber u.a. zur Aufspaltung in verschiedene Programmzweige. Ein größeres Hindernis gegen eine Aufspaltung können aber Markenrechte für bestimmten Quellcode darstellen, da nur bestimmter Quellcode mit einem Logo bzw. einem Markennamen verbunden werden darf.

Im Weiteren wird daher genauer auf Koordinationsformen als Normen bzw. Institutionen der sozialen Bewegungen eingegangen, die auch im KDE-Projekt wirken können.

- Eine (gruppenorientierte) *Kooperation* als Koordinationsform zwischen den Einheiten der sozialen Open Source-Bewegung basiert wie in anderen sozialen Bewegungen auf gemeinsamen normativen Grundlagen und Ansichten. Eine starke soziale Solidarität mit direkten, egalitären Beziehungen als symbolische, intrinsische Ressource der Beteiligung wie im KDE-Projekt spielt eine dominierende Rolle (vgl. Della Porta, Diani 1999: 140ff.). Dabei zeigt sich der Zusammenhalt der virtuellen sozialen Bewegung besonders in den virtuellen, bewegungsweiten Diskursen z.B. über Wertorientierungen auf den Kommunikationsplattformen, wobei gegenseitiges Vertrauen beim Austausch von Informationen herrscht und jeder teilnehmen darf (s. Kap. 5.1.; vgl. Della Porta, Diani 1999: 122f.). Kooperationen konnten zwischen einzelnen sozialen Bewegungen, aber auch ihren Gruppen bzw. Organisationen wie in der Open Source-Bewegung festgestellt werden. Eine kooperative Haltung und die gemeinsamen Werte in der Open Source-Bewegung zeigen sich z.B. in der Übergabe von Strafverfolgungen von Lizenzübertretungen an die FSF. Zusätzlich helfen sich die Bewegungsbeteiligten durch die gemeinsame, informale Sanktionierung von Lizenzübertretungen z.B. von Unternehmen über schlechte Publicity in den Medien (s.o.; vgl. Della Porta, Diani 1999; McAdam et al. 1988; Rucht et al. 1997). Ein weiteres Beispiel für eine kooperative Beziehung innerhalb der Open Source-Bewegung ist, dass die FSF ihre Server für das Herunterladen von Open Source-Software-Releases oder eine Liste von Mitarbeiter suchenden Projekten zur Verfügung stellt und sie darüber indirekt auch Aufmerksamkeit erhält (Scacchi, Elliott 2004). Zudem ist eine wichtige Norm in Open Source-Bewegung und KDE-Projekt die Softwarelizenz GPL, die zusammen mit Markenrechten eine Kooperation u.a. im Open Source-Projekt unterstützt (s.o.). Die GPL-Lizenz wurde vom KDE-Projekt imitiert, wobei sie auch durch die FSF kontrolliert und sanktioniert wird.

Außerdem entstanden und bestehen freiwillige selbstorganisierte Gruppen mit hohem kooperativem Anteil innerhalb der sozialen Bewegungen der 1960er bis 1980er Jahre, wobei Gemeinschaft und kooperatives Arbeiten gerade in der Open Source-Bewegung bzw. der FSF als wichtige Werte angesehen werden. Die sozialen Bewegungen sind in ihrer Struktur eher diffus, bestehen aber modular aus selbstorganisierten Kleingruppen auf der zivilgesellschaftlichen Gemeinschaftsebene (vgl. Kap. 5; Scacchi, Elliott 2004; Schattenhofer 1992: 14ff.; Bernart 1999: 290; Della Porta, Diani 1999: 119; Claessens 1999: 157). Damit kann die soziale Bewegung die Entwicklung von selbstorganisierten Gruppen aus dem

Gemeinschaftssystem stützen, zu denen auch die Open Source-Projekte wie das KDE-Projekt gehören. Außerdem zeigt eine Zugehörigkeit zur gleichen sozialen Bewegung eine Beziehung und den Zusammenhalt zwischen den Projekten auf.

- *Hierarchie/Macht* lässt sich in einer sozialen Bewegung auch durch viele dezentrale, charismatische Führer mit begrenzter Gefolgschaft oder eine Kerngruppe nachweisen, die aus Personen mit hohem sozialem Status in der Bewegung besteht (vgl. Della Porta, Diani 1999: 140f.). Hierarchie/Macht zeigt sich in der Bewegung außerdem in Form einer informellen, aber auch formalen Organisation oder Gruppe, die die Führung innerhalb der Bewegung übernimmt. Innerhalb dieser Organisation oder Gruppe in der Bewegung kann eine formale, aber auch informale Hierarchie/Macht auftreten (vgl. Rucht et al. 1997: 153f.). Führer oder Kerngruppe informaler oder formaler Art treten auch im KDE-Projekt auf. Hierarchie/Macht in einer sozialen Bewegung, aber besonders der Open Source-Bewegung beruht oftmals auf einer informellen Hierarchie mit fluktuierender Postenbesetzung und/oder mit einer charismatischen Führerschaft, die z.T. auf die gesamte soziale Bewegung ausstrahlt, aber keine Rolle im KDE-Projekt spielt (Della Porta, Diani 1999: 24ff.; 137ff.; Scacchi, Elliott 2004). Im Open Source-Projekt KDE scheinen die dominierenden Normen der Wissenschaft und der außeruniversitären Profession eine charismatische Führerschaft zu verhindern bzw. abzuschwächen.

In der Open Source-Bewegung geht zwar eine informale Hierarchie von der Vertretungsorganisation FSF aus, da die Vertretungsorganisation die Überwachung der Softwarelizenz GPL für das KDE-Projekt aber auch für andere Projekte sowie die Außendarstellung gegenüber den Medien und der Politik übernimmt (s.o. bzw. [www.fsf.org](http://www.fsf.org)). Außerdem hat sich Expertenwissen auf Seiten der Vertretungsorganisation gebildet, so dass diese bei politischen und lizenzrechtlichen Themen größeren Einfluss hat. Dies entspricht einer Zentrum-Peripherie-Struktur der zentralen Aktiven gegenüber einem mobilisierbarem Milieu (Della Porta, Diani 1999: 24ff.), die auch innerhalb des KDE-Projekts, aber auch außerhalb zwischen Mitgliedern und projektexternen Programmierern vorgefunden werden kann.

- Ein nachgeordneter *Wettbewerb* kann wie in sozialen Bewegungen, zwischen Open Source-Projekten als Einheiten einer sozialen Bewegung um potenzielle Mitarbeiter und Meinungsführerschaft in der öffentlichen Debatte/Meinung existieren (Della Porta, Diani 1999: 124ff.; McAdam et al. 1988; Rucht et al. 1997). Gleiches gilt auch für das KDE-Projekt. Es besteht die Norm, dass keine Mitarbeiter abgeworben werden oder keine aggressive Werbung um potenzielle Mitglieder erfolgt, weswegen Kooperation dominiert. Die Projekte werben eher über eine hohe Qualität ihres Produkts, die sie beibehalten müssen, sonst werden diese Projekte Nutzer und Entwickler durch Inaktivität verlieren oder ein Zwillingprojekt zweigt sich auf der Basis desselben Quellcodes ab. Bei diesem Wettbewerb kann ein Projekt entweder verschwinden oder es wendet sich spezifischen Nischen zu, wie NetBSD, FreeBSD und OpenBSD (Working Group 2000: 15f.). Als Beispiel für ein Abzweigen (Forking) kann die Konkurrenz zwischen Gnome- und KDE-Projekt bei der Entwicklung von grafischen Benutzeroberflächen gesehen werden. Gnome gründete sich als Folge eines Streits zwischen Free Software Foundation und KDE-Projekt über die Lizenzbedingungen der für das KDE-Projekt grundlegenden, von der Firma Trolltech zur Verfügung gestellten Bibliothek. Das jeweils andere Projekt wird beobachtet und Ideen bzw. neue Entwicklungen von Softwarefunktionen manchmal aufgegriffen bzw. imitiert. Gelegentlich folgt aus einem projektinternen Wettbewerb z.B. wegen mangelnder Konformität von Teilgruppen mit der Softwarelizenz oder mit gemeinsamen Normen durch eine Abspaltung ein projektexterner. Diese Abspaltung kann für das Projekt negativ sein, da

Personal dem Projekt abhanden kommt, aber positiv für die Bewegung, da ein „produktiver“ Wettbewerb entsteht.<sup>243</sup>

### Politiksystem mit juristischem System und Staat/staatlicher Verwaltung

Im Politiksystem sind nicht nur die sozialen Bewegungen, sondern auch der Staat/die staatliche Verwaltung und das juristische System angesiedelt, die normalerweise Organisationen und individuelle Akteure sanktionieren. Dabei kann unter Sanktionen nicht nur die juristische Verurteilung mit Strafe, sondern auch der Entzug von (materiellen oder finanziellen) Ressourcen verstanden werden, auf die das KDE-Projekt mit ihrer Softwareentwicklung angewiesen ist (vgl. Kap. 6.2.1.2.).

Anders als bei anderen sozialen Bewegungen bekommt das KDE-Projekt Geld nicht vom Staat, aus Mitgliedsbeiträgen oder dem Verkauf von Gütern/Dienstleistungen, sondern es handelt sich ausschließlich um finanzielle oder materielle Spenden besonders von Universitäten oder Unternehmen<sup>244</sup>, aber manchmal auch von Privatleuten bzw. KDE-Mitgliedern (vgl. <http://www.kde.org/support/sponsoring/thanks.htm>; Brand, Holtgrewe 2007; Rucht et al. 1997: 106, 122ff.; Della Porta, Diani 1999: 140).

Die staatlichen Stellen und ihre Verwaltung unterstützen das Projekt nicht unmittelbar mit Geld, sondern finanzieren über projektnahe Unternehmen spezifische Softwareentwicklungen für Open Source-Teilprojekte, die eher dem Gemeinwohl oder dem Staat dienen (vgl. Projekt Ägypten beim KDE-Projekt<sup>245</sup>; Hoegner 2006). Auch wurde kaum ein Softwareentwickler in der staatlichen Verwaltung oder im juristischen System sozialisiert, weswegen die sehr hierarchischen Verwaltungsstrukturen nicht auf das KDE-Projekt oder andere Projekte übertragen werden konnten.

Aus Verwaltung und juristischem System werden also keine sozialisierten Normen übernommen. Stattdessen haben sich wie bei selbstorganisierten Gruppen und bei sozialen Bewegungen beim untersuchten KDE-Projekt zur rechtlich abgesicherten Kanalisierung von Sponsorengeldern formale inkorporierte Organisationen wie Non-Profit-Vereine bzw. –Stiftungen (KDE e.V. bzw. KDE League) entwickelt, die u.a. auch als klassische Non-Profit-Organisationen die Außenbeziehungen des Projekts managen (vgl. Klein 2001: 713f.; McAdam et al. 1988: 722; Rucht et al. 1997: 112ff., 152f.; Schattenhofer 1992: 120; Nave-Herz 1993: 80f.).<sup>246</sup> Der KDE-Verein verwaltet und verteilt nicht nur die Sponsorengelder, sondern besitzt wie die Free Software Foundation und Open Source-Initiative alle wichtigen Rechte (Copyright-, Markenrechte) zur Verteidigung des gemeinsamen Produkts gegenüber proprietärer Aneignung vor allem durch Unternehmen (vgl. O'Mahony 2003a, b).<sup>247</sup> Damit ist, wie oben im Zusammenhang mit der Open Source-Bewegung dargestellt

---

<sup>243</sup> Der Erfolg des Open Source-Modells bzw. der Open Source-Bewegung kann auch im Ineinandergreifen von Kooperation als Basis mit Konkurrenzelementen gesehen werden. Einerseits treibt Wettbewerb zwischen den Open Source-Projekten neue Entwicklungen voran, die andererseits über Kooperation zwischen den Open Source-Projekten ausgetauscht werden (Working Group 2000: 15f.). Bei Unternehmen hingegen ist dies nicht der Fall.

<sup>244</sup> Die wichtigsten Spender sind die Universitäten Kaiserslautern, Bonn und Tübingen sowie bei den Unternehmen der Distributeur SUSE und die Firma Trolltech (s. Kap. 5; <http://www.kde.org/support/sponsoring/thanks.htm>).

<sup>245</sup> s. <http://www.gnupg.org/aegypten/index.de.html>

<sup>246</sup> Zudem kann weder die informale soziale Open Source-Bewegung noch das informale Open Source-Projekt KDE wie in anderen sozialen Bewegungen auf die formalen Strukturen eines machtvollen Verbündeten z.B. einer politischen Partei zurückgreifen und hat daher eine eigene formale Organisation gegründet (vgl. Della Porta, Diani 1999: 155f.).

<sup>247</sup> Auch Professionen bilden formale Interessensorganisationen oder Vereine aus, um in der Politik gehört zu werden bzw. rechtliche Macht zur Selbstorganisation vom Staat übertragen zu bekommen (Freid-

wurde, der Verein mit einer möglichen juristischen Sanktionierung von Mitarbeitern, Unternehmen oder projektexternen Nutzern verbunden, kann aber auch selbst durch die Open Source-Bewegung auf juristischem Wege sanktioniert werden.

Eine Nicht-Formalisierung würde zu einem geringen Spendenaufkommen führen, was auch als eine Art Sanktion auf das Open Source-Projekt wirkt. Zusätzlich zwingt wahrscheinlich die wachsende Gruppengröße das KDE-Projekt, aber auch andere Open Source-Projekte, sich zu formalisieren, d.h. eine formale, geregelte Verwaltung als Verein aufzubauen und sich als Großgruppe zu organisieren. Eine gewachsene Gruppengröße benötigt Außenkontakte und Ressourcen aus der gesellschaftlichen Umwelt, da sonst die Infrastruktur für diese Größe nicht zur Verfügung stünde (vgl. Claessens 1983, 1999). Die spendenden Organisationen erwarten dabei vertrauenswürdige Strukturen. Die Struktur des KDE-Vereins ähnelt daher einerseits den Strukturen des politischen Systems bzw. der Demokratie (vgl. Heun 1983), andererseits auch mit ihrer Non-Profit-Ausrichtung den Zielen der sozialen Bewegung.

Die formale Struktur des Vereins bzw. die Ausarbeitung der Satzung des KDE-Vereins wurde anhand der Vorgaben des Vereinsrechts aus dem juristischen System imitiert, wobei „falsche“ Satzungsparagraphen durch das juristische System sanktioniert werden können (vgl. Hasse, Krücken 1999: 16). Im Vereinsrecht sind formale Posten mit hoher Entscheidungsmacht (z.B. Vorstand) und ein demokratisches Wahlrecht bei Postenverteilung und der Themenabstimmung vorgeschrieben (vgl. Jütting et al. 2003; Klein 2001; <http://ev.kde.org/statutes-de.htm>). Diese Struktur hat aber keinen Einfluss auf die Struktur der Softwareentwicklung im KDE-Projekt, wie beim Debian- oder Gnome-Projekt beobachtet (s.u.). Diese formalen Institutionen sind nur bedingt im KDE-Projekt vorzufinden, was mit der geringen Abhängigkeit von den Spenden zusammenhängen kann. Die Vereinsstruktur unterstützt die Einrichtung von Posten wie Projektleiterposten, die auch in selbstorganisierten Gruppen auftreten. Vereinsposten spielen jedoch bei den Aktivitäten im KDE-Projekt lediglich eine geringe oder gar keine Rolle, da sie keinen Einfluss über eine übergreifende Letztentscheidungsmacht auf die Softwareentwicklung haben.

Da der Verein nur zur Kanalisierung von Spenden eingerichtet wurde, wobei formale, sanktionierbare Normen imitiert wurden, hat die Vereinsstruktur also nur einen mittelbaren Einfluss auf die Softwareentwicklung im KDE-Projekt. Damit besteht nur ein geringer Institutionalisierungsgrad. Weitere Anreize aus dem juristischen System oder vom Staat/der staatlichen Verwaltung bestehen kaum, die eine Übernahme in die (informale) Aktivitätsstruktur rechtfertigen könnten. Zusätzlich stehen die übernommenen institutionellen Strukturen der außeruniversitären Profession, wissenschaftlichen Disziplinen und selbstorganisierten Gruppe gegen die Struktur des Vereins, da die Sozialisation von realen und potenziellen Mitarbeitern wichtiger ist als die finanziellen und materiellen Anreize, die die Infrastruktur am Laufen halten.

### Wirtschaftssystem

Das Wirtschaftssystem kann auch Einfluss auf das KDE-Projekt nehmen. Das KDE-Projekt hat zwar Kontakt mit Unternehmen, doch werden keine Unternehmensstrukturen übernommen. Das Projekt soll, wie oben beschrieben, eher legitim gegenüber Universitäten als gegenüber Unternehmen erscheinen, da einerseits die finanziellen oder materiellen

---

son 1994: 68, 174). In den Disziplinen ist dies ähnlich: An großen Großforschungseinrichtungen der Physik bestehen Gutachtergruppen zur Verteilung der Arbeitszeit und ein Komitee mit formaler Entscheidungsgewalt zur finanziellen Ressourcenverteilung, an dem alle Teilinstitute der Forschungseinrichtung teilnehmen (Knorr-Cetina 2002: 254ff.).

Spenden als Ressource einen nachrangigen Stellenwert haben. Andererseits kommen die Spenden nicht nur von Unternehmen, sondern besonders von Hochschulen (s.o.). Manche Unternehmen spenden aber hin und wieder einen relativ großen Betrag (s. <http://www.kde.org/support/sponsoring/thanks.htm>, 23.11.2005). Von diesen Unternehmen spenden einige Großunternehmen aus der Softwarebranche oftmals nur einmalig bzw. sporadisch, haben aber sonst kaum Kontakt mit dem KDE-Projekt.

Das Personal der potenziellen Mitarbeiter kommt eher nicht aus Unternehmen, sondern viele Mitarbeiter sind Studenten. Einige der KDE-Entwickler arbeiten zwar in mittelständischen Firmen bis Großunternehmen und können mit dem Segen des Unternehmens während ihrer Arbeitszeit für das KDE-Projekt arbeiten. Nur wenige Softwareentwickler vom KDE-Projekt werden jedoch Vollzeit für ihre Erwerbsarbeit im KDE-Projekt von Firmen finanziert. Die erwerbsarbeitenden KDE-Entwickler trennen überwiegend strikt zwischen selbstbestimmtem Hobby mit Entspannungseffekt und Erwerbsarbeit, wobei sie keine Strukturen ähnlich der Erwerbsarbeit in Unternehmen im Open Source-Projekt wünschen (s. Kap. 5.2.; Holtgrewe 2004a). Diese Unterschiede werden wie in selbstorganisierten Gruppen ignoriert, um keine formale Hierarchie zwischen den erwerbsarbeitenden und anderen KDE-Entwicklern entstehen zu lassen (vgl. Schattenhofer 1992: 120f.). Damit haben die Normen der Unternehmen kaum Einfluss im Projekt, auch da die Institutionen der selbstorganisierten Gruppen dagegen stehen.

Die KDE-Softwareentwickler haben zudem mit softwareentwickelnden Unternehmen Kontakt, die am ehesten mit kooperativen Professionsunternehmen verglichen werden können. In diesen Unternehmen dominiert zwar Hierarchie/Macht der Posten, es bestehen aber auch Kooperationsstrukturen, die nicht in Konflikt mit anderen Institutionen stehen (s.u., vgl. z.B. Lazega 2001 für Rechtsanwaltsfirmen).

Ein weiterer Grund kann in der Abfolge der Sozialisation liegen: die KDE-Projektmitarbeiter werden vorher in das KDE-Projekt sozialisiert und arbeiten später in Unternehmen. Die KDE-Projektmitarbeit ist dabei ein Lebenslaufphänomen, wobei viele nach Eintritt in ein Unternehmen nicht mehr im KDE-Projekt mitarbeiten (s.o.; vgl. Brand, Holtgrewe 2007). Zwar sind die KDE-Entwickler gegenüber Unternehmen relativ aufgeschlossen, haben aber trotzdem wenig direkten, offiziellen Kontakt zu diesen (vgl. Brucherseifer 2004).

Ein weiterer Punkt, warum die Struktur der Unternehmen nicht übernommen werden, kann auch in der Struktur der und der Vorstellungswelt in Unternehmen liegen. In diesen Unternehmen dominiert zwar Hierarchie/Macht, aber kooperative Elemente in Form der Projektarbeit treten vermehrt auf. Die Projektarbeit in Softwareunternehmen ähnelt den institutionellen Open Source-Projektstrukturen, wobei die KDE-Mitarbeiter am ehesten Kontakt zu Softwareentwicklern in Softwareunternehmen haben. Doch erfolgt die Projektarbeit in Softwareunternehmen in formalen „Zwangsgruppen“ innerhalb einer Unternehmenshierarchie und können daher aus den oben genannten Gründen nicht einfach übertragen werden. Zusätzlich existiert in der Managementliteratur eine Vorstellung von Projekten mit kooperativen Elementen, die in den letzten Jahren u.a. ausgehend von sozialen Bewegungen verstärkt Einzug in die Unternehmen besonders der Softwarebranche nahmen (vgl. Boltanski, Chiapello 2003; Welz, Ortman 1992). Damit unterstützt die Ideologie von Netzwerken/Kooperation in der Managementliteratur die Legitimation von Open Source-Projekten bei Unternehmen. Dies wirkt sich aber nur in der Form aus, dass Unternehmen das Projekt über materielle oder finanzielle Spenden sponsern.

Da nun die institutionellen Strukturen der Unternehmen keinen Einzug in das KDE-Projekt halten, stützt die in Unternehmen bedeutendere Hierarchie Hierarchie/Macht im KDE-Projekt nicht, so dass diese nicht bedeutender als Kooperation ist.

Umgekehrt werden die Institutionen des Open Source-Projekts KDE in die Wirtschaft getragen. Das Open Source-Projekt KDE ist dabei nicht aus einem Unternehmen entstanden, sondern umgekehrt: viele Kleinunternehmen entstanden aus dem KDE-Projekt und haben die gleichen Werte wie die soziale Bewegung internalisiert. Ähnliche kooperative Strukturen sind in Genossenschaften und selbstverwalteten Betrieben zu finden, die aus anderen sozialen Bewegungen entstanden (vgl. Atmaca 2002; Heider, Mevissen 1991).

Zudem spielen kooperative Elemente wegen der starken Verknüpfung von Produktentwicklung und Unternehmensentwicklung in kleinen Unternehmen, besonders der Softwarebranche und als Neugründungen, generell eine große Rolle. Die Arbeit macht oftmals den Unternehmenssoftwareentwicklern Spaß und sie sehen diese ähnlich den Open Source-Entwicklern als Professionalisierung ihres Hobbys. Konsensbildung und Vertrauen haben in Kleinunternehmen eine hohe Bedeutung, sind aber anders als beim Open Source-Projekt vom „Eigentümerunternehmer“ mit dominierender Hierarchie abhängig (Nett, Wulf 2005: 156f.; Baukrowitz, Boes 2002: 21; Hirsch-Kreinsen 2003: 18; Kotthoff, Reindl 1990). Die Gründung von Kleinunternehmen aus der sozialen Bewegung oder einem Open Source-Projekt führt nicht zu einer Übernahme einer dominierenden Hierarchie. Vielmehr ist die Art der Kooperation wie im Open Source-Projekt wichtig.

Obwohl das KDE-Projekt kaum Organisationsstrukturen aus der wirtschaftlichen Umwelt übernimmt, ist Windows bzw. Microsoft eine Umwelt des Projekts. Mit Microsoft wird um die Endnutzer und unterstützenden Unternehmen konkurriert (s. Kap. 5.1.4.4., 4.2.; Leiteritz 2002). Diese Konkurrenz lässt das KDE-Projekt die Designstrukturen und Ideen von Office-/Browser-Programmen der Microsoft Software imitieren und die Legitimität bei den Endnutzern erhöhen. Diese Orientierung an der Technik von Windows soll Endnutzer und damit potenzielle Softwareentwickler den Umstieg auf die eigene KDE-Software erleichtern und die Rekrutierungszahlen erhöhen.

#### Andere Open Source-Projekte als Bezugsgruppe

Das KDE-Projekt unterhält auch Kontakte zu anderen Open Source-Projekten, die ebenfalls Softwareprodukte herstellen. Das Projekt fühlt sich dabei anderen Open Source-Projekten zugehörig, die damit eine „Bezugsgruppe“ für das KDE-Projekt darstellen. In dieser Bezugsgruppe verfolgt das KDE-Projekt, das Linux-Projekt mit seinem Open Source-Betriebssystem und das Debian-Projekt mit seiner Open Source-Distribution das gemeinsame Interesse, eine miteinander kompatible Softwaredistribution herauszugeben. Es besteht auch ein sporadischer Kontakt mit Kooperation zwischen den Projekten z.B. über gemeinsame Standardisierungsbemühungen im Gnome- und KDE-Projekt für die fehlerfreie Softwareentwicklung auf Linux-Basis (s. [www.freedesktop.org](http://www.freedesktop.org)<sup>248</sup>).

Dabei werden KDE-Projekt und andere Open Source-Projekte aus den gleichen gesellschaftlichen Umwelten wie soziale Open Source-Bewegung, selbstorganisierte Gruppen, Wissenschafts- und Professionssystem beeinflusst. Die Open Source-Projekte ähneln bzw. gleichen sich in groben Zügen in ihrer Kombination der Koordinationsformen, wobei eine ähnliche IuK-technische Infrastruktur auf ähnliche Koordinationsformen hindeutet. Das Mailinglisten- und Fehlerberichtssystem ermöglicht beispielsweise eine gleichgestellte, dominierende Kooperation zwischen den Mitarbeitern, aber auch Endnutzern. Mit diesen Infrastrukturen sind zugleich neue Infrastruktur-Leitungspositionen entstanden, die hierarchischen Elementen gleichkommen. Reputationshierarchien und ein latenter, unsichtbarer Wettbewerb sind außerdem zu beobachten, was auf die oben diskutierte glei-

---

<sup>248</sup> s. [http://wiki.linuxquestions.org/index.php?title=KDE\\_v\\_Gnome\\_history.htm](http://wiki.linuxquestions.org/index.php?title=KDE_v_Gnome_history.htm)

che Herkunft zurückgeführt werden kann (s. Kap. 5.4 für eine genaue Darstellung der wissenschaftlichen Diskussion um Koordinationsformen in den Open Source-Projekten).

Die Zugehörigkeit zu einer sozialen Bewegung und der gemeinsame Zusammenhang mit anderen Open Source-Projekten der Bezugsgruppe lässt auch Mitglieder des KDE-Projekts Verfahren und Koordinationsstrukturen von anderen Open Source-Projekten *imitieren*. Während des sporadischen Kontakts zwischen Vertretern der Projekte werden auch neue Ideen ausgetauscht. Das KDE-Projekt imitierte z.B. die Erstellung von Releases.<sup>249</sup> Eine Sozialisation in anderen Open Source-Projekten erfolgte nur bei den Gründern, die Erfahrungen aus anderen Projekten mitbrachten (s.o.). Eine weitere Sozialisation hat wegen des geringen Wechsels zwischen den Projekten kaum eine Bedeutung. (vgl. Tuomi 2001). Eine Imitation zwischen den Projekten konzentriert sich aber hauptsächlich auf technische Lösungen, wie Quellcode, fertige Programme oder technische Ideen, und kaum auf andere soziale Strukturen neben der Institution Release. Dies wird dadurch unterstützt, dass die verschiedenen Open Source-Projekte unterschiedliche Software produzieren, die jeweils eine eigene technische Funktion für die Open Source-Bewegung darstellen und die ggf. übernommen werden können (s. Kap 5.1.). Das KDE-Projekt hat technische Werkzeuge als Ressourcen übernommen, d.h. IuK-technische Infrastruktur- oder Produktionssoftware wie das Versionsmanagement-, Fehlerbericht-, Homepage- und Mailsystem, an die eigenen Bedürfnisse anpasst und manchmal auch weiterentwickelt.<sup>250</sup> Umgekehrt wurden Produktionswerkzeuge wie Texteditoren, Compiler, etc. und das Softwareprodukt des KDE-Projekts von anderen Projekten bezogen.

Die Imitation von institutionellen Strukturen aus anderen Projekten hat aber ihre Grenzen (vgl. Walgenbach 2002: 171ff.). Das KDE-Projekt übernimmt besonders technische Lösungen, da es Anerkennung über das Softwareprodukt erhält und die technischen Lösungen direkt zu einer einfacheren Softwareentwicklung beitragen. Anerkennung ist mit der Software bzw. dem Quellcode zudem die dritt wichtigste Ressource, weswegen eher die Normen aus der gesellschaftlichen Umwelt das KDE-Projekt bestimmen, die mit der Ressource Personal mit Sozialisation verbunden sind. Das KDE-Projekt hat dagegen bisher keine formalisierten, demokratischen Wahlverfahren oder formalisierten Aufnahmeverfahren als Ausprägungen von Koordinationsformen aus anderen Open Source-Projekten imitiert. Die hier auftretende Imitation, die durch die Sozialisation als Strukturangleichungsmechanismus dominiert wird, ist dabei der drittstärkste Strukturangleichungsmechanismus.

Die Koordinationsformen als Institutionen werden nicht aus dem Institutionengeflecht der anderen Open Source-Projekte übernommen, da beispielsweise das Gnome-, Debian- oder Linux-Projekt - wie oben angedeutet - leicht andere gesellschaftliche Umwelteinflüsse haben. Über die Gründe für diese Nicht-Übernahme wird an dieser Stelle nachgedacht, indem das KDE-Projekt mit dem Gnome-, Debian- oder Linux-Projekt wegen eines anderen Forschungsfokusses eher grob verglichen wird. Weitere vergleichende Studien können diese Zusammenhänge vertiefen.

---

<sup>249</sup> Die Gründer hatten bei ihrem vorigen Projekt bisher keinen Release erlebt (der erste Release von Lyx kam erst mehrere Monate nach dem ersten Release von KDE heraus; s. <http://www.mail-archive.com/lyx-announce@lists.lyx.org/msg00003.html>), kannten Releases aber von dem mit Lyx verbundenen Projekt Latex (s. <http://www.latex-project.org/ltnews/ltnews01.pdf>).

<sup>250</sup> Gerade das Versionsmanagementsystem Concurrent Versioning System (CVS) wurde von einem Open Source-Projekt entwickelt und von anderen, darunter KDE übernommen (Bauer, Pizka 2003: 3, 5).

Die mangelnde Imitation von nicht-technischen Institutionen scheint besonders mit einer formalen inkorporierten Organisation zusammenzuhängen, d.h. einem formalen Verein/einer Foundation, die in vielen Open Source-Projekten bestehen (s.o.; vgl. O'Mahony 2003a, b). Der höhere Einfluss des Vereins/der Foundation auf den Ablauf der Softwareentwicklung im Debian- und Gnome-Projekt gegenüber dem KDE-Projekt kann daran liegen, dass das Debian- und das Gnome-Projekt beispielsweise in den USA, das KDE-Projekt dagegen in Deutschland gegründet wurden. Dominieren im KDE-Projekt die Deutschen, dominieren in den beiden anderen Projekten US-amerikanische Entwickler und ihre demokratischen Werte, was mit ein Grund für die bedeutendere formale Organisation Verein/Foundation sein könnte.<sup>251</sup>

Der größere Einfluss des Vereins/der Foundation führt wiederum zu einer höheren Bedeutung von Wettbewerbsmechanismen. Beim Debian- und Gnome-Projekt<sup>252</sup> spielen daher häufig formalisierte Mehrheitswahlen eine Rolle, in denen projektumfassende, wichtige Themen und Postenbesetzungen abgestimmt werden. Der stärkere Einfluss von Hierarchie/Macht durch die formale Organisation Verein/Foundation zeigt sich u.a. beim stark formalisierten Aufnahmeverfahren im Debian- gegenüber dem KDE-Projekt. Dabei kann die Betonung von formalem Aufnahmeverfahren, Schlichtungskomitee<sup>253</sup> und gewähltem Gesamtprojektleiter auf die soziale Nähe des Open Source-Distributors Debian zu den Endnutzern bzw. Unternehmen und ihrem Bedürfnis nach fehlerfreier Software zurückgeführt werden (vgl. Holtgrewe 2005). Gegenüber dem Debian- ist beim KDE-Projekt die Abhängigkeit von privaten Endnutzern und Unternehmen als Nutzern nicht so groß (vgl. Holtgrewe 2005), da die kommerziellen Distributionsunternehmen und das Debian-Projekt direkte Ansprechpartner für Endnutzer sind.<sup>254</sup> Auch beim Gnome-Projekt beeinflusst die Gnome-Foundation gegenüber dem KDE-Projekt über eine Hierarchie den Ablauf der Softwareentwicklung stark.<sup>255</sup> Gnome wird dabei stärker als das KDE-Projekt durch Unternehmen dominiert, die eine formale Organisation einer Foundation als Ansprechpartner benötigen. Gnome ist stark mit den Unternehmen Novell (vormals Ximian) und Red Hat, aber auch anderen Unternehmen wie HP, IBM, Sun etc. verbunden, die über ihre Sponsorentätigkeit und die Entwicklung von wichtigen Teilprogrammen des Projekts einen besonderen Einfluss auf das Gnome-Projekt haben (vgl. <http://foundation.gnome.org/>; <http://de.wikipedia.org/wiki/GNOME>; German 2003).<sup>256</sup> Außerdem arbeiten viele mehr angestellte Softwareentwickler aus den Unternehmen als Open-Source-Entwickler an Teilbereichen der Software mit und werden von dem Unternehmen dafür bezahlt als im KDE-

---

<sup>251</sup> s. Geschichte von Debian (<http://www.debian.org/doc/manuals/project-history/index.de.html>) und von Gnome (<http://primates.ximian.com/~miguel/gnome-history.html>) sowie die Weltkarte von Debian (<http://www.debian.org/devel/developers.loc.de.html>).

<sup>252</sup> s. z.B. „<http://www.debian.org/vote/>“ und „<http://foundation.gnome.org/referenda/2004-10/>“

<sup>253</sup> Beim Debian-Projekt wurde im Gegensatz zum KDE-Projekt durch eine formale Festlegung auf der Homepage die Schlichtung von technischen Konflikten einem mehrköpfigen technischen Komitee und von persönlichen und gefühlsintensiven Konflikten dem gewählten Gesamtprojektleiter zugewiesen (vgl. <http://www.debian.org/devel/tech-ctte>).

<sup>254</sup> Beim KDE-Projekt ist u.a. auch daher der Verein nicht zentral für den Ablauf der Softwareentwicklung.

<sup>255</sup> Die Gnome-Foundation bestimmt die Mitglieder von Komitees wie dem Release Komitee, Infrastruktur Komitee (Sysadmin committee) etc., die Entscheidungen über Teilfunktionen der Softwareentwicklung fällen. Außerdem kontrolliert die Gnome-Foundation den formalisierten Diskussionsablauf bei strittigen Fragen bzgl. der Quellcode-Übernahme in einen Release (vgl. German 2003; <http://www.debian.org/vote/>; <http://foundation.gnome.org/membership.html>).

<sup>256</sup> Der große Einfluss der Unternehmen zeigt sich auch bei der Wahl des Vorstands der Foundation in 2005, bei der einige Personen aus den wichtigen Unternehmen (z.B. Novell, Red Hat etc.) kandidierten und z.T. hineingewählt wurden (s. <http://www.pro-linux.de/news/2005/9014.html>; [http://foundation.gnome.org/vote/results.php?election\\_id=2](http://foundation.gnome.org/vote/results.php?election_id=2)). Beim KDE-Projekt ist dies bis auf wenige Ausnahmen kaum der Fall.

Projekt. Dies gilt auch für das Debian-Projekt, was obiges Argument der direkten Beziehung von Unternehmen und Projekt untermauert.<sup>257</sup> Diese Softwareentwickler der Unternehmen benötigen formale Regeln, wie sie sie aus Unternehmen kennen, auch um die Interessen zwischen Unternehmensentwicklern und freiwilligen Softwareentwicklern ohne projektumfassende Konflikte ausgleichen zu können (vgl. German 2003). Dagegen ist beim KDE-Projekt der Einfluss der Universitäten durch ihre Spenden und der Studenten hoch. Zugleich entwickeln sehr wenige Unternehmen Softwareteile im KDE-Projekt.

Beim Linux-Projekt besteht eine stärkere Hierarchie als im KDE-Projekt, die aber nicht auf eine formale Organisation Verein/Foundation zurückgeführt werden kann. Wegweisende Planungen und Entscheidungen über die Aufnahme von Quellcode werden seit langem unverändert bei Linux durch einen wohlmeinenden Diktator und seine Lieutenants gefällt (vgl. Kuwabara 2000; Grassmuck 2002: 237f.). Die stärkere Hierarchie könnte an der Abhängigkeit anderer Open Source-Projekte, aber auch von Unternehmen, von der Fehlerfreiheit des zentralen Betriebssystems liegen.

Kurze Zusammenfassung aller Einflüsse von Akteuren aus den Gesellschaftsbereichen auf das KDE-Projekt

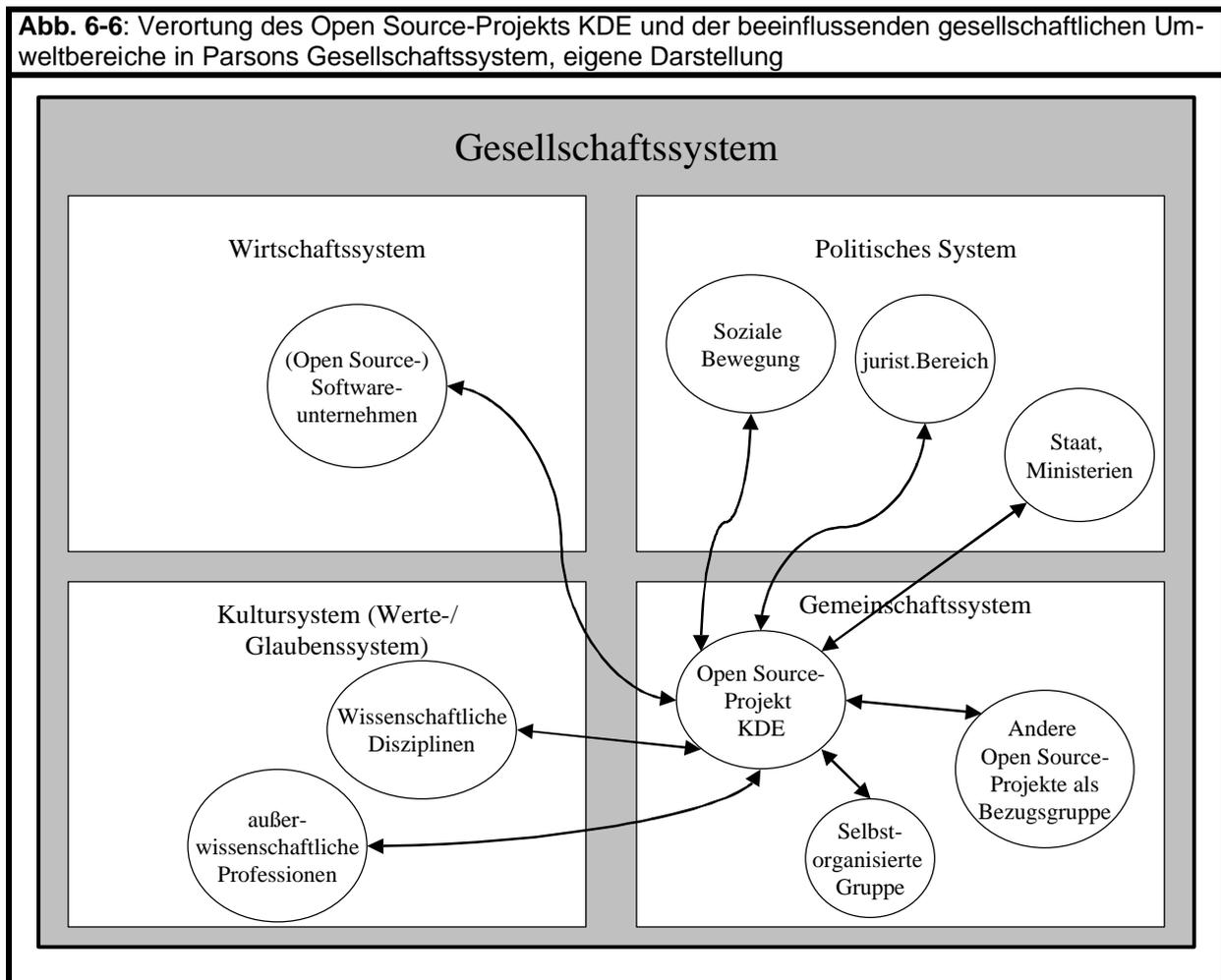
Bisher wurde untersucht, wie die Koordinationsformen als Institutionen in den relevanten Akteuren der Gesellschaftsbereiche die Institutionen im KDE-Projekt vermutlich bestimmen. Die zentralen Strukturangleichungsmechanismen legen mit den Ressourcen fest, aus welchen Gesellschaftsbereichen und welchen Akteuren Koordinationsformen vermutlich ins KDE-Projekt integriert werden. Im Mittelpunkt steht, welche ähnlichen Koordinationsformen aus den verschiedenen Akteuren der Gesellschaftsbereiche sich gegenseitig unterstützen und somit diese Koordinationsformen stark institutionalisieren können.

Die Akteure aus den Gesellschaftsbereichen, die die Institutionen im KDE-Projekt beeinflussen, können für ein besseres Verständnis in Parsons Gesellschaftssystem verortet werden. Im Gesellschaftssystem werden das KDE-Projekt, selbstorganisierte Gruppen ebenso wie außeruniversitäre Professionen, wissenschaftliche Disziplinen, soziale Open Source-Bewegung, Unternehmen, Staat/Ministerien sowie der juristische Bereich verortet (s. Abb. 6-6).

---

<sup>257</sup> s. z.B. <http://www.debian.org/partners/index.de.html>

**Abb. 6-6:** Verortung des Open Source-Projekts KDE und der beeinflussenden gesellschaftlichen Umweltbereiche in Parsons Gesellschaftssystem, eigene Darstellung



Von den relevanten, zentralen Akteuren in den Gesellschaftsbereichen wie selbstorganisierte Gruppen im Gemeinschaftssystem, außeruniversitäre Professionen und grundlagenwissenschaftliche Disziplinen Mathematik und Physik im Kultursystem sowie soziale Open Source-Bewegung im Politiksystem werden alle drei Koordinationsformen über die Sozialisation und Ressource Personal im Institutionengeflecht des Open Source-Projekts KDE institutionalisiert, d.h. ins KDE-Projekt kommen eher in diesen Bereichen sozialisierte Studenten. Zusätzlich kann aus der verwendeten Literatur vermutet werden, dass ähnlich wie im KDE-Projekt in diesen Gesellschaftsbereichen Kooperation im Zentrum steht, Hierarchie/Macht als Reputationshierarchie und Posten nachgeordnet sowie Wettbewerb auf Reputation ausgerichtet und nicht nennenswert ist. Die spezielle Softwarelizenz aus der sozialen Open Source-Bewegung gepaart mit Markenrechten unterstützt Kooperation durch die juristische Sanktionierung einer proprietären Aneignung der Software.

Eine Sanktionierung erfolgt nicht über eine juristische Strafe, sondern über den Entzug von Spenden. Materielle und finanzielle Spenden als nachrangige Ressource kommen kaum vom Staat/staatlicher Verwaltung, sondern eher von Hochschulen und Unternehmen. Koordinationsformen aus Staat/staatlicher Verwaltung werden deswegen nicht übernommen. Auch eine dominierende Hierarchie wird aus Unternehmen wegen der sporadischen Kontakte und z.T. sporadischen Spenden kaum übernommen. Andere Einflüsse auf die Softwareentwicklung aus dem Wirtschaftssystem werden wahrscheinlich durch spezifische Kontakte zu Unternehmen verhindert, die ähnliche Strukturen und Vorstellungen von Projektarbeit mit kooperativen Elementen wie in Open Source-Projekten haben.

Zur Kanalisierung von Spenden wurde im KDE-Projekt ein Verein gegründet, der aber keinen großen Einfluss auf die laufende Softwareentwicklung und somit auf die Koordinationsformen hat, d.h. an diesem Punkt besteht eine Entkoppelung zwischen dem Verein als Formalstruktur und der Aktivitätsstruktur. Die zentrale, formale Stellung des Vereins/der Foundation für den Arbeitsprozess der Softwareentwicklung in anderen Open Source-Projekten wird nicht imitiert, da Spenden vor allem von Universitäten kommen, die dem Wissenschaftssystem angehören. Zudem ist der Umfang der Spenden und der Kontakt zu Unternehmen beim KDE-Projekt geringer als beim Gnome- oder Debian-Projekt, bei denen der Verein/die Foundation eine zentrale Stellung auch beim Ablauf der Softwareentwicklung hat.

Außerdem imitiert das KDE-Projekt aus seiner Bezugsgruppe aus anderen Open Source-Projekten nur technische Lösungen wie Quellcode, fertige Programme oder technische Ideen und bezieht nur Anerkennung als Ressource aus dieser Bezugsgruppe. Die Bezugsgruppe kann das KDE-Projekt nicht formal, sondern nur bedingt informal sanktionieren und eine Sozialisation des Personals in anderen Projekten findet kaum statt.

#### *6.2.2.2. Analyse der Koordinationsformen in den Teilprozessen der Softwareentwicklung*

In diesem Unterkapitel wird der Einfluss der gesellschaftlichen Akteure auf die Koordinationsformen in den Teilprozessen des KDE-Projekts als zweiter Zwischenschritt untersucht. Der vermutliche Einfluss auf die Bedeutung der Koordinationsformen wird anhand der Institutionen aus den wichtigsten relevanten Akteuren der Gesellschaftsbereiche festgestellt. Die Institutionen dieser Akteure, die sozialisiert wurden und mit der Ressource Personal verbunden sind, dominieren im Open Source-Projekt. Die wichtigsten relevanten Akteure sind, wie im vorangegangenen Unterkapitel dargelegt, selbstorganisierte Gruppen, außeruniversitäre Professionen und wissenschaftliche Disziplinen sowie soziale Bewegungen.

Da der Arbeitsprozess im Mittelpunkt dieser Arbeit steht, in der Literatur zu diesen Akteuren kein solcher Arbeitsprozess untersucht wurde und die Koordinationsformen in der Literatur den Teilprozessen zugeordnet wurden, wird im Folgenden eher eine grobe Analyse durchgeführt. Es kann angenommen werden, dass durch Sozialisation in den wichtigsten relevanten Akteuren die Institutionen dieser Akteure übernommen werden. Die Bedeutung der Koordinationsformen im KDE-Projekt wird daher anhand ähnlicher Institutionen der wichtigsten Akteure durch Assoziationen analysiert. Die Institutionen bzw. Koordinationsformen eines Akteurs, die in ähnlichen Situationen wie dem untersuchten Teilprozess auftreten, beeinflussen einen Teilprozess. Die Koordinationsformen verschiedener Akteure aus den Gesellschaftsbereichen stützen sich dabei gegenseitig und verkeilen diese im Teilprozess. Damit kann eine hohe Bedeutung einer Koordinationsform für einen Teilprozess durch eine hohe Anzahl von Institutionen der Akteure bestimmt werden oder diese Koordinationsform eine ähnliche Bedeutung im Institutionengeflecht des Akteurs hat. Die Bedeutung einer Koordinationsform in einem Teilprozess im KDE-Projekt steigt also mit der Anzahl der Akteure aus einem Gesellschaftsbereich, in denen eine solche Koordinationsform mit ähnlicher Bedeutung in einer ähnlichen Situation vorhanden ist.

Im Folgenden wird also auf dieser konkreteren Ebene zwischen den Institutionen der relevanten Akteure und den Institutionen im Open Source-Projekt KDE in den Teilprozessen Eintritt/Rekrutierung, Arbeitsplanung (Zieldefinition, Arbeitsverteilung von Arbeitspaketen mit detaillierter Entwicklungsplanung, Arbeitsverteilung von Posten), Leistungserstellung mit detaillierter Entwicklungsplanung und Arbeitszusammenführung verglichen und vermutliche Ähnlichkeiten gesucht. Da sich eher allgemeine Aussagen über Koordinationsformen in der Literatur finden lassen, besteht jedoch eine letzte Unsicherheit, ob sich so alle Koordinationsformen der Akteure vollständig und befriedigend feststellen lassen.

## 1. Eintritt/Rekrutierung

Die Institutionen des Eintritts bzw. der Rekrutierung werden durch außeruniversitäre Profession und wissenschaftliche Disziplinen, soziale Bewegung sowie vor allem durch die selbstorganisierte Gruppe bestimmt. Der Einstieg in einen Beruf, eine soziale Bewegung oder selbstorganisierte Gruppe erfolgt ähnlich zum Open Source-Projekt weitgehend graduell. Dabei ist bei einer Profession eine eindeutiger Entscheidung von den Eintretenden üblich.<sup>258</sup> Anders als in Professionen, bei denen die formale Sozialisation durch Basislehrveranstaltungen an Hochschulen oder spätere Trainings-/Seminarprogramme eine große Rolle spielt, existiert beim KDE-Projekt keine formale Sozialisation. Im Gegenteil ist die informale Sozialisation wie in selbstorganisierten Gruppen oder bei sozialen Bewegungen bestimmend, da diese Akteure die direkten Rekrutierungspools für potenzielle Mitarbeiter darstellen (Trice, Beyer 1993: 194, 196f.; Levine, Moreland 1995: 428f.; Schattenhofer 1992: 141f.). Eine informale Sozialisation erfolgt bei sozialen Bewegungen und ähnlich bei selbstorganisierten Gruppen über eine bestehende soziale Beziehung zu einem Bewegungsmittglied und eine Sozialisierung über informale Subgruppen, was einen graduellen Eintritt unterstützt (Della Porta, Diani 1999: 112ff., 122; Oliver, Marwell 1992: 266; McAdam et al. 1988: 707f.). Dies gilt auch für das KDE-Projekt.<sup>259</sup>

Beim Eintritt spielt in selbstorganisierten Gruppen, aber auch in der sozialen Bewegung Sympathie als Motivationsbasis bei potenziellen Mitgliedern eine Rolle. Die soziale Bewegung rekrutiert, wie in anderen neuen sozialen Bewegungen potenzielle Mitglieder auch mit einem bestimmten gesellschaftlichen Status, wobei Sympathie auch eine Rolle spielt. Die potenziellen Mitglieder sind wie beim Open Source-Projekt vor allem Jugendliche bzw. junge Erwachsene der Mittelklasse in einer Transitions- und Orientierungsphase, d.h. oftmals Studenten, mit einer intrinsischen Motivation bzgl. eines Mobilisierungsthemas (vgl. Brand 1998; Della Porta, Diani 1999: 115; Rucht et al. 1997: 103f., 149f.; Sader 1991: 84ff.; Levine, Moreland 1995: 428f.; Abbott 1988: 198). Austritte bei selbstorganisierten Gruppen, vor allem bei einem Generationenwechsel zwischen Neu- und Altmitgliedern, erfolgen durch ein Wegbleiben oder abnehmender Beteiligung, wie auch beim KDE-Projekt zu erkennen ist (Schattenhofer 1992: 109ff.).

- Beim Eintritt von potenziellen Mitarbeitern ins KDE-Projekt dominiert *Hierarchie/Macht*, da wie in selbstorganisierten Großgruppen vor allem ein Posten über den Eintritt entscheidet (vgl. Claessens 1983). In der Wissenschaft besteht eine ähnliche Hierarchie/Macht in Form eines Lehrer-Schüler-Verhältnisses oder eines wissenschaftlichen Torwächters, wobei die Eintrittspforte in jedem Fall durch eine Prüfung kontrolliert wird. Die Torwächter bei Profession und Wissenschaft erreichen, ähnlich zu den Subprojektleitern beim KDE-Projekt, durch ihre Beobachtung von potenziellen Mitarbeitern eine Personalselektion (vgl. Freidson 2001: 72ff., 86, 94; Bühl 1974: 87f.). Dies korrespondiert mit den verschärften Auslesekriterien von potenziellen Mitarbeitern bei Großgruppen, um die Funktionsfähigkeit der Gruppe zu erhalten (Claessens 1983: 491).<sup>260</sup> Diese Institutionen wurden nicht nur in den selbstorganisierten Gruppen und sozialen Bewegungen als bedeutendste Koordinati-

---

<sup>258</sup> Der Eintritt in eine bürokratische Organisation unterscheidet sich davon stark, da eine zeitliche Bindung und ein genau abgegrenzter Anfang besteht.

<sup>259</sup> Die informale Sozialisation erfolgte bei KDE über Bekanntschaft auf Konferenzen oder Freundschaften, den Wechsel von Lyx zu KDE und den Bezug des potenziellen Mitglieds zu einem KDE-Subprojekt mit entsprechender Software (s.o.; vgl. Brucherseifer 2004).

<sup>260</sup> Dabei findet eine Abwägung des Torwächters zwischen dem Anspruch der Profession und der Wissenschaft statt, die sich in der Begrenzung der Professionsmitglieder für einen Arbeitsmarktschutz (Profession) und der Offenheit der Wissenschaft für wissenschaftliche Talente (Norm des Universalismus) und für Wissensaustausch darstellt (Freidson 2001: 94; Bühl 1974: 107).

onsform sozialisiert, sondern auch in der Wissenschaft. Mit dieser hohen Anzahl an beeinflussenden Institutionen ist Hierarchie/Macht sehr stark institutionalisiert.

- *Wettbewerb* hängt mit Hierarchie/Macht im Institutionengeflecht der beeinflussenden Akteure zusammen und wird mit diesem auch sozialisiert. Ein Vergleich des Wissens, der Leistungsbereitschaft und der Ähnlichkeit des potenziellen Mitglieds mit Mitgliedern der selbstorganisierten Gruppe, der sozialen Bewegung bzw. mit anderen publizierenden Wissenschaftlern wird durch den Torwächter durchgeführt. Dieser Wettbewerb ähnelt dem Vorgehen im Open Source-Projekt KDE (s.o.). Da eine formale Stellenbesetzung wie in der Wissenschaft oder Profession nicht erfolgt, herrscht eher ein informaler Wettbewerb wie in selbstorganisierten Gruppen und sozialen Bewegungen. Eine mittlere Anzahl von beeinflussenden Institutionen führt zu einer mittleren Bedeutung.

- Bei selbstorganisierten Gruppen und sozialen Bewegungen treten wie im KDE-Projekt potenzielle Mitglieder über eine bestehende soziale Beziehung zu einem bisherigen Mitglied in Kontakt mit der Gruppe oder Bewegung (s.o.). Sympathie und damit Vertrauen spielt eine Rolle, kann aber die Aufnahme in die Gruppe oder Bewegung nicht garantieren. Außerdem ist im KDE-Projekt eine ähnliche reziproke *Kooperation* wie zwischen Lehrer und Schüler/Professor und Student bzw. zwischen Wissenschaftler und einem „Torwächter der Wissenschaft“ als Posten wie Zeitschriftenherausgeber, wissenschaftliche Gutachter, Doktorväter und Rezensenten bedingt zu erkennen: Der Neuling muss sich zwar selbst einarbeiten (Selbstsozialisation), bekommt aber bei Problemen und Fragen vom Posten wie Projektleiter hin und wieder Hilfe. Ähnlich ist es bei der sozialen Bewegung, bei der die Gruppe die Art der Hilfe beim Eintritt bestimmt (Levine, Moreland 1995: 428f.; Wildavsky 1986; Della Porta, Diani 1999: 112ff.; vgl. Brand, Holtgrewe 2007). Damit ist Kooperation zwar eine Grundlage beim Eintritt und wird bei selbstorganisierten Gruppen und sozialen Bewegungen sozialisiert, ist jedoch die am wenigsten bedeutende Koordinationsform in der Gruppe und Bewegung.

### 2.1. Arbeitsplanung: Zieldefinition

Auch die Erstellung einer Zieldefinition bei der Arbeitsplanung wird durch die Institutionen von selbstorganisierten Gruppen, außeruniversitären Professionen, wissenschaftlichen Disziplinen und sozialen Bewegungen bestimmt.

- *Kooperation*, die in diesen Akteuren dominiert, kann daher auch im Open Source-Projekt dominieren. Besonders viele beeinflussende Institutionen aus den Akteuren können zu einer hohen Bedeutung dieser Koordinationsform führen. Bei einer sozialen Bewegung werden zukünftige, abstrakte Ziele, aber auch Argumentationen bzgl. Werten durch Treffen, aber auch Publikationsorgane wie bewegungsinterne Zeitschriften sowie andere Kommunikationstechnologien wie Telefon oder Internet, ausgetauscht bzw. diskutiert, wobei sich jedes Bewegungsmitglied beteiligen kann. Zieldefinitionsbezogene Kooperationen werden bei sozialen Bewegungen, Professionen und selbstorganisierten Gruppen wie auch in der Mathematik durch einen gemeinsamen Grundkonsens getragen, was sich auch im KDE-Projekt nachweisen lässt (Kliment 1998; McAdam et al. 1988: 715; Heintz 2000a: 220). Eine Grenze der Sozialisation in der sozialen Bewegung besteht im KDE-Projekt in der Abwesenheit eines politisch-ideologischen Diskurses inklusive Konflikten und dem Vorhandensein einer kooperativen Diskurskultur mit fachlich-technischer Argumentation. Dies ist auf die sich verstärkende, gleiche Sozialisation in den gesellschaftlichen Bereichen der formalwissenschaftlich-physikalischen Professionen bzw. Disziplinen und der selbstorganisierten Gruppe zurückzuführen (vgl. Brand, Holtgrewe 2007; Schattenhofer 1992).

Bei Großgruppen wird auf (Großgruppen-)Sitzungen kooperativ diskutiert, die eine Gruppe zweiter Ordnung auf der vertikal obersten Ebene darstellen und entweder aus besonders engagierten Personen oder aus Vertretern der Basisgruppen eines Großgruppenverbandes bestehen (vgl. Claessens 1999; Rauch 1983). Ähnlich ist es in der Wissenschaft, wobei in der Forschungseinrichtung der Physik wissenschaftliche Gremien ohne Beteiligung aller Wissenschaftler, aber mit Vertretern aller Gruppen die beantragten Experimente nach sachlichen Argumenten begutachten und grobe Ziele festgelegt werden. Zwischen Forschergruppen wird aber auch kooperiert, indem Wissen über den Forschungsgegenstand ausgetauscht wird (s.o.; Knorr-Cetina 2002: 244, 255ff.; Laudel 2001).<sup>261</sup>

Im KDE-Projekt existieren diese Diskurse in Form von Mailinglisten, wobei textbasierte Mailinglisten eine Mischung der Charakteristika von Publikationen und Sitzungen/Treffen haben. Einerseits sind über Publikationen im Gegensatz zu Sitzungen prinzipiell alle Beteiligten erreichbar, andererseits ist bei Sitzungen im Gegensatz zur Publikation eine sofortige Entgegnung möglich. In den Mailinglisten besteht zudem als kooperatives Element eine allgemeine Sachorientierung, wie in den Professionen und Wissenschaftsdisziplinen Ingenieurwissenschaften, Mathematik und Physik üblich. Bei Entscheidungen wird also die technisch vernünftigste Lösung gesucht, wobei eine gruppenorientierte Kooperation zwischen den Wissenschaftlern und wie im KDE-Projekt besteht. Es wird in Gremien versucht, eine gemeinsame Lösung zu finden, wobei jeder beitragen kann (s.o.; vgl. Knorr-Cetina 2002: 255ff., 275, 282ff.; Kelty 2001; Poulsen 2001; Heintz 2000a: 153). Auch Großgruppensitzungen sind eher kognitiv-aufgabenorientiert und weniger affektiv ausgerichtet (vgl. Rauch 1983). In einer selbstorganisierten, freiwilligen Gruppe werden üblicherweise allgemeine Beiträge zum entweder aktions- oder identitätsorientierten Ziel der Gruppe von jedem Mitglied akzeptiert und eine gemeinsam getragene Entscheidung bzw. ein finaler Konsens gesucht (vgl. Schattenhofer 1992, 1997; Levine, Moreland 1995: 449).<sup>262</sup> Besonders die informalen Leitungspositionen oder Personen mit Reputation (innerer Kreis) kooperieren in selbstorganisierten Gruppen wie im KDE-Projekt besonders miteinander (vgl. Schattenhofer 1992: 136). Entscheidungen erfolgen mit z.T. stillschweigender Billigung der anderen Mitglieder, wie in den Mailinglisten des KDE-Projekts oder anderen rein textbasierten selbstorganisierten virtuellen Gruppen (vgl. Götzenburger, Löger 2000).

- Eine nachgeordnete *Hierarchie/Macht* besteht, da weniger Institutionen aus der gesellschaftlichen Umwelt diesen Teilprozess beeinflussen bzw. diese Koordinationsform in den relevanten Akteuren auch von nachgeordneter Bedeutung ist. In den einzelnen Akteuren bestehen Gruppen zweiter Ordnung, in denen „mächtige“ Personen über strategische Belange auch der weiteren Kleingruppen diskutieren. Wie in sozialen Bewegungen sind im KDE-Projekt beispielsweise einzelne informale oder formale Gruppen wie Softwareentwickler, Übersetzer, Dokumentare etc. mit einem vertikalen und modularen Aufbau aus Kleingruppen anzutreffen (vgl. Della Porta, Diani 1999: 140ff., 124ff.; McAdam et al. 1988: 711; Claessens 1999). Auch Großforschungseinrichtungen der Physik bestehen aus kleineren, formalen Forschergruppen, die es an Großforschungseinrichtungen der Mathematik nur als informale Ad hoc-Gruppen gibt (vgl. Knorr-Cetina 2002; Heintz 2000a). Diese formalen und informalen Forschergruppen bzw. Organisationen der sozialen Bewegung wer-

---

<sup>261</sup> Zwar fließen die Informationen und das Wissen in dem physikalischen Großexperiment und beim KDE-Projekt von den konkreten Arbeitsgruppen zu den abstrakten Entscheidungsgremien und werden besonders an Plenartreffen/Konferenzen weitergegeben, doch besteht durch die Mailinglisten beim KDE-Projekt zudem eine direkte Einbettung der Subprojekte in den allgemeinen Diskurs (Knorr-Cetina 1992: 246).

<sup>262</sup> Ein finaler Konsens wird gesucht, da sonst die Gruppe auseinandergehen würde (vgl. Schattenhofer 1992: 108; Levine, Moreland 1995).

den wie im KDE-Projekt über Diskurse mit grundlegenden Überzeugungen und speziellen übergreifenden Gruppen (zweiter Ordnung) miteinander verbunden. Einer Großgruppensitzung dieser Gruppe zweiter Ordnung ähneln die Diskussionen in einer zentralen KDE-Mailingliste.<sup>263</sup> Dabei wird in einer Großgruppensitzung der Diskurs von wenigen zentralen Beteiligten aufrechterhalten und gleicht eher dem Diskurs in einer Kleingruppe.

Zudem beeinflussen Reputation und Expertenwissen als hierarchische Elemente die Diskussion im KDE-Projekt wie bei Wissenschaftlern, Professionsmitgliedern und in selbstorganisierten, freiwilligen Gruppen. Dabei spielen die Posten eher eine nachgeordnete Rolle (vgl. Knorr-Cetina 2002; Freidson 2001: 144f.; Schattenhofer 1992: 138, 1997; Heintz 2000a: 159 Fußnote 27; 180). In sozialen Bewegungen beeinflussen manchmal sogenannte Oligarchien die Diskussion stark in ihrem Sinne (Della Porta, Diani 1999: 161). Auch bei Großgruppensitzungen entsteht, wie in gesamtprojektbezogenen Mailinglisten des KDE-Projekts, eine Hierarchie/Macht zwischen mehreren (meist vier bis acht) Hauptsprechern, den gelegentlich Aktiven und den vielen Passiven (vgl. Rauch 1983). Dabei beeinflusst der innere Kern auch bei anderen virtuellen selbstorganisierten Gruppen über IuK-technische Sanktionen (wie technischen Gruppenausschluss) und textbasiert-soziale, „verbale“ Sanktionen (wie Beschimpfen, Ironie, Ignorieren) sowie eher negative Sanktionierung den Gruppenprozess (vgl. Döring, Schestag 2000; Götzenburger, Löger 2000; Hoffmann 2000; Thiedecke 2000a, b).

- Der *Wettbewerb* hat im KDE-Projekt eine geringe Bedeutung, da bei der Forschungseinrichtung der Physikdisziplin ein eher unbedeutender Wettbewerb um finanzielle oder zeitliche Ressourcen zwischen den Forschergruppen, in der Mathematik eher um Urheberchaft herrscht. Wettbewerb wird wie im KDE-Projekt durch eine Sachorientierung kaschiert (Knorr-Cetina 2002: 276ff.; Heintz 2000a: 158ff., 203). Eine Ausschreibung bzgl. Publikationen, die mit dem Quellcode im KDE-Projekt korrespondieren, findet in der Wissenschaft eher selten statt. In Großgruppensitzungen bzw. in gesamtprojektbezogenen Mailinglisten des KDE-Projekts ist ein Wettbewerb um Arbeitszeitressourcen bzw. eine zukünftige Entwicklungsrichtung mit einem Wettbewerb zwischen den Beteiligten um Aufmerksamkeit mit unterschiedlichen Rangpositionen, zwischen verschiedenen Hauptrichtungen der Argumentation und ihren Fürsprechern oder zwischen verschiedenen Koalitionen in einer Gruppe bei Entscheidungen verbunden (vgl. Rauch 1983; Levine, Moreland 1995: 440f., 448). Die Beeinflussung bzw. Monopolisierung der Diskussion durch eine Minderheit z.B. von Oligarchen kann im schlimmsten Fall zu einem Auseinanderbrechen einer Gruppe oder einer sozialen Bewegung führen (Della Porta, Diani 1999: 161). Durch die Fiktion der Gleichberechtigung werden solche Konflikte in selbstorganisierten Gruppen wie im KDE-Projekt tabuisiert und eher unterbunden, um ein Auseinanderbrechen zu verhindern, was mit einem geringen Meinungswettbewerb zusammenhängt (Schattenhofer 1992: 125ff., 1997). Wettbewerb wird dabei von wenigen Institutionen aus der gesellschaftlichen Umwelt unterstützt.

## 2.2. Arbeitsplanung: Arbeitsverteilung von Aufgabenbereichen/-paketen mit detaillierter Entwicklungsplanung

Die Arbeitsverteilung von Aufgabenbereichen/-paketen mit detaillierter Entwicklungsplanung bei der Arbeitsplanung ist ähnlich strukturiert wie die Zieldefinition. Auch hier bestimmen die Institutionen der selbstorganisierten Gruppen, außeruniversitären Professi-

---

<sup>263</sup> Beim KDE-Projekt hat sich daher eine vertikale Unterscheidung zwischen Mailinglisten etabliert, wobei eine Mailingliste für alle projektexternen und –internen Softwareentwickler (kde-devel) und eine Mailingliste für ausgesuchte KDE-Mitglieder (kde-core-devel; mit Personen mit Reputation, aber z.T. auch Vertretern bzw. Leitern der Subprojekte) einer Großgruppensitzung ähneln.

onen, wissenschaftlichen Disziplinen und sozialen Bewegungen das KDE-Projekt. Da sich diese Institutionen kaum von den oben genannten Institutionen der Akteure bei der Zieldefinition unterscheiden, wird an dieser Stelle nur auf Besonderheiten der Arbeitsplanung eingegangen.

Detailliert geplant wird im KDE-Projekt wie in den einzelnen diversen Organisationen und Gruppen der sozialen Bewegung, in den Kleingruppen von selbstorganisierten Großgruppen und den Forschergruppen, wobei auch eine Sachorientierung besteht. Politische, wertbezogene Diskussionen wie bei der Zieldefinition werden kaum geführt, da eher das konkrete Vorgehen im Mittelpunkt steht. Die Struktur der Diskussionen unterscheidet sich bzgl. Organisationen und Gruppen der sozialen Bewegung, der verschiedenen Gruppen des Großforschungsprojekts und der selbstorganisierten Gruppen wenig (vgl. Della Porta, Diani 1999: 119ff.; Claessens 1999; Schattenhofer 1992: 122f.; Knorr-Cetina 2002: 242ff.; Poulsen 2001). Gerade in den selbstorganisierten (Klein-)Gruppen, dem KDE-Projekt, aber auch in sozialen Bewegungen oder der Forschungseinrichtung der Mathematik, werden freiwillig spezielle Aufgaben übernommen, wobei sich diese Rollenzuweisung verfestigen kann. In der wissenschaftlichen Disziplin Physik werden Entscheidungen wie im KDE-Projekt oftmals gegenstandsnahe von den beteiligten Experten getroffen. Gegenseitige Hilfe und damit reziproke *Kooperation* spielt im KDE-Projekt aber auch in den Forschergruppen der Physikdisziplin, und den selbstorganisierten Gruppen in Form von untergeordneten, spontanen, informellen Arbeitsgruppen oder zwischen zwei Gruppenmitgliedern eine Rolle. Damit wird diese durch viele Institutionen der gesellschaftlichen Akteure verkehrt. Weniger bedeutende *Hierarchien* durch Reputation oder Posten sind zudem in selbstorganisierten (Klein-)Gruppen, Forschergruppen und sozialen Bewegungen zu finden. Diese wenigen Institutionen führen zu einem nachgeordneten Stellenwert. Ein *Wettbewerb* um Aufgaben existiert im KDE-Projekt, in selbstorganisierten Gruppen, in Gruppen der sozialen Bewegung oder Forschergruppen laut Literatur nicht, was zu einer geringen Bedeutung dieser Koordinationsform führt (vgl. Schattenhofer 1992: 121ff.; 154ff.; Oliver, Marwell 1992; Della Porta, Diani 1999; Knorr-Cetina 2002: 242; 252ff.; Poulsen 2001; Laudel 2001; Heintz 2000a, b).

### 2.3. Arbeitsplanung: Arbeitsverteilung durch Postenbesetzung

Die Arbeitsverteilung durch Postenbesetzung bei der Arbeitsplanung wird vermutlich hauptsächlich durch die selbstorganisierten Gruppen und die soziale Bewegung bestimmt, da aus der Literatur abgeleitet die beiden anderen Akteure weniger Einfluss haben. Die Postenbesetzung im KDE-Projekt erfolgt wie in selbstorganisierten Gruppen freiwillig, aus eigenem Antrieb (intrinsische Motivation), um die Arbeit der Gruppe zu vereinfachen. Diese Posten haben aber einen eher informalen Charakter und beruhen auf einer Arbeitsteilung bzw. Rollenübernahme in der Gruppe (Schattenhofer 1992: 134ff.). Die neue Postenbesetzung geht aber wie in selbstorganisierten Gruppen von dem bisherigen Posteninhaber als *hierarchisches* Element aus, wobei der Nachfolger aus dem Kern der Gruppe rekrutiert wird und den Posten freiwillig übernimmt (Schattenhofer 1992: 154ff.). Auch in den sozialen Bewegungen wird auf diese Weise rekrutiert (Della Porta, Diani 1999: 112ff; Oliver, Marwell 1992: 266). In selbstorganisierten Gruppen werden die Nachfolger aus der Gruppe unter stillschweigender Billigung der restlichen Mitglieder eingesetzt, wobei dies in den Mailinglisten des KDE-Projekts aber auch anderen rein textbasierten selbstorganisierten virtuellen Gruppen auftritt<sup>264</sup> (Schattenhofer 1992: 154ff., 1997: 178; Sader 1991: 290; Götzenburger, Löger 2000). Ähnlich ist in der Wissenschaft die informale Vergabe von

---

<sup>264</sup> Demgegenüber geht streng geregelt in Selbsthilfegruppen die Besetzung der Führungsposition reihum (Schattenhofer 1992, 1997).

Arbeitsstellen durch den Projektleiter und die Vergabe nach der Reputationshöhe. Dabei tritt zwar auch eine formale Postenbesetzung vor allem an Universitäten auf, wobei Posten nicht durch Posteninhaber sondern von Gremien vergeben werden (vgl. Knorr-Cetina 2002: 238f., 266). Die formale Postenbesetzung hat aber manchmal nur den Stellenwert einer Formalität, wie beim KDE-Projekt. Die formalen Strukturen der Postenbesetzung haben kaum einen Einfluss, da sie im Gegensatz zu den informalen, sozialisierten Strukturen der sozialen Bewegungen, der selbstorganisierten Gruppen und der Wissenschaft in Form der Forschungseinrichtung der Mathematik stehen, in der es eher informale als formale Posten gibt (vgl. Heintz 2000a). Die hohe Bedeutung von Hierarchie/Macht wird durch die hohe Anzahl an korrespondierenden Institutionen aus der gesellschaftlichen Umwelt erreicht.

Die nachgeordnete *Kooperation* basiert auf der sozialisierten Kooperation (wie stillschweigender Billigung neuer Posteninhaber oder von Entscheidungen) in selbstorganisierter Gruppe, Wissenschaft, außeruniversitärer Profession und sozialer Open Source-Bewegung wie in Arbeitsverteilung von Aufgabenbereichen/-paketen (s.o.). Da in der Literatur über die Akteure keine weiteren Institutionen kooperativer Postenbesetzung genannt werden, hat die Kooperation eine mittlere Bedeutung.

*Wettbewerb* ist unbedeutend, da in der dominierenden selbstorganisierten Gruppe, aber auch den Kleingruppen in Wissenschaft und sozialer Bewegung laut Literatur kein Wettbewerb vorherrscht. Zudem kommen die Postennachfolger aus dem KDE-Projekt. Bei der Postenbesetzung besteht jedoch keine freiwillige Prüfung, kein Abgabe oder Publikations-, Demonstrations-/Protesttermin wie in den Wissenschaften, Professionen oder sozialen Bewegungen.

### 3. Leistungserstellung mit detaillierter Entwicklungsplanung

Die Institutionen der selbstorganisierten Gruppen, außeruniversitären Professionen, wissenschaftlichen Disziplinen und sozialen Bewegungen beeinflussen die Leistungserstellung mit detaillierter Entwicklungsplanung im KDE-Projekt stark.

- Die Leistungserstellung mit detaillierter Entwicklungsplanung wird beim KDE-Projekt und bei technisch-physikalischer Disziplin bzw. Profession durch *Kooperation* dominiert, da besonders viele Institutionen aus der gesellschaftlichen Umwelt diese Koordinationsform verketten. Beispielsweise bestehen in Communities Of Practice oder Epistemic Communities wie im KDE-Projekt Solidarität, eine gegenseitige Hilfe und Wissensaustausch wegen ähnlicher Probleme (Klatschzirkel). In diesem Teilprozess ist der wissenschaftliche Versuchs-und-Irrtums-Prozess am ehesten anzutreffen. Gerade durch das Internet mit den Mailinglisten wird die reziproke, aber auch gruppenorientierte Kooperation in Wissenschaft und Profession unterstützt. Bei konkreten, praxisbezogenen Spezialproblemen, die einzelne Professionsmitglieder nicht allein lösen können, werden Experten oder andere Professionsmitglieder wie im KDE-Projekt angesprochen. In der Disziplin Mathematik wird zur Fehlerkontrolle erwartet, dass noch nicht publikationsreife Texte und Ideen ausgetauscht werden (s.o.; Freidson 2001: 101; Knorr-Cetina 2002: 280; Laudel 2001; Heintz 2000a: 181, 202; Hinner 2003; Abbott 1988: 40f.). Auch in sozialen Bewegungen tritt wie im KDE-Projekt unaufgeforderte und gegenseitige Hilfe auf, nicht nur innerhalb einer Bewegung oder einzelner Gruppen, sondern auch zwischen verschiedenen sozialen Bewegungen (Della Porta, Diani 1999; McAdam et al. 1988; Rucht et al. 1997). Dabei unterstützt der wissenschaftliche Ethos eine Kooperation im KDE-Projekt (vgl. Taubert 2006).

- Eine nachgeordnete *Hierarchie/Macht* zeigt sich in den technisch-physikalischen Professionen oder Disziplinen, sozialen Bewegungen wie im KDE-Projekt, indem Projektleiter, Experten oder informale Leiter sich eher einbringen, Ideen generieren oder Fragen beantworten. Dabei erfolgt eine Kontrolle durch den formalen bzw. informalen

worten. Dabei erfolgt eine Kontrolle durch den formalen bzw. informalen Projektleiter, aber auch eine gegenseitige, kollegiale „Kontrolle“. In selbstorganisierten Gruppen werden Personen, die nicht das Gruppenziel (beim KDE-Projekt die Softwareentwicklung) verfolgen, gegen informale Normen verstoßen oder zu unterschiedlich in sozialem Status oder Meinung sind, wie im KDE-Projekt durch Ignorieren oder Ausschluss sanktioniert (vgl. Schattenhofer 1992: 108ff.). Damit stützen weniger Institutionen Hierarchie/Macht als bei der Kooperation, was die mittlere Bedeutung dieser Koordinationsform bestätigt.

Der eher unbedeutende *Wettbewerb* um Reputation zeigt sich in formalwissenschaftlich-physikalischen Professionen oder Disziplinen, sozialen Bewegungen aber auch selbstorganisierten Gruppen durch ständige Leistung und Anwesenheit wie im KDE-Projekt. Die ständige Leistung und Anwesenheit tritt bei der Wissensgenerierung der Professionen und Disziplinen, bei der Protestvorbereitung bei sozialen Bewegungen und Verwirklichung des Gruppenziels bei selbstorganisierten Gruppen auf (s.o., vgl. Knorr-Cetina 2002; Laudel 2001; Heintz 2000a, b; Schattenhofer 1992, 1997; Della Porta, Diani 1999; McAdam et al. 1988).

#### 4. Arbeitszusammenführung

Ein Release mit Releaseplan prägt die Arbeitszusammenführung beim KDE-Projekt. Dabei sind nicht nur die sozialisierten Institutionen aus den selbstorganisierten Gruppen, außeruniversitären Professionen, Disziplinen sowie sozialen Open Source-Bewegungen anzutreffen. In diesem Teilprozess bestimmen auch Institutionen von anderen Akteuren wie Unternehmen und Bezugsgruppe die Arbeitszusammenführung.

- Release, Releasekoordinator und Releaseplan sind *hierarchische* Elemente und wurden u.a. von anderen Open Source-Projekten der Bezugsgruppe imitiert. Das KDE-Projekt führte den Release mit der Auslieferung der fehlerfreien, funktionsfähigen Software und Endtests ein, da zudem eine Sanktionierung durch einen versiegenden Zustrom aller drei Ressourcen (Spenden, Personal und Anerkennung) durch Unternehmen und Endnutzer erfolgen kann.

Als Vorbild für KDE-Projekt und Bezugsgruppe kann dabei das Verfahren des Wasserfallmodells gelten, das häufig in der kommerziellen Softwareentwicklung in Unternehmen verwendet wird. Bei der Softwareentwicklung in Softwareunternehmen werden abgestufte Zwischen- und Endtests durchgeführt, wobei im KDE-Projekt der Release dem Endtest ähnelt (s. Kap. 2.3.; vgl. Welz, Ortmann 1992).

Weiterhin wirken im KDE-Projekt an dieser Stelle auch Institutionen der ingenieurwissenschaftlichen Disziplin, da durch einen offiziellen Release eine funktionstüchtige, erfolgreiche Software entsteht. Gerade bei den Ingenieurwissenschaften sind solche Endtests zentral (vgl. Banse et al. 2006; Welz, Ortmann 1992).<sup>265</sup> Das Vorgehen in den Ingenieurwissenschaften ähnelt dabei dem Releaseplan und dem Wasserfallmodell (vgl. Glock 1997; Ropohl 2004; Banse et al. 2006).

In diesem Teilprozess kann eine Nähe zur Bürokratisierung in Unternehmen festgestellt werden, da die Releasekoordinatoren offiziell auf der Homepage und in den Mailinglisten

---

<sup>265</sup> Eine dezentrale Erstellung bei gleichzeitiger zentraler Koordination (u.a. auch bei sozialen Bewegungen) dürfte auch bei Großgruppen kurz vor größeren Aktionen vorzufinden sein, ist aber in der soziologischen Literatur kaum und in der sozialpsychologischen Literatur nicht dargestellt worden. Rauch (1983: 256) konstatierte Anfang der 1980er Jahre einen Mangel an Forschung zu Großgruppen, was sich seitdem auch nicht geändert hat (s. Literaturverzeichnis von Claessens 1999). Die Überschneidungen zwischen den Definitionen von Organisationen und Großgruppen spielen wohl eine Rolle.

verkündet werden und damit einen formalen Charakter bekommen.<sup>266</sup> Im Unternehmen kontrolliert ein Vorgesetzter bis hin zum Unternehmenschef die Durchführung, wobei Ähnlichkeiten zum Releasekoordinator bestehen (vgl. Welz, Ortman 1992).

Diese zusätzlichen Einflüsse anderer Akteure unterstützen die höhere Bedeutung von Hierarchie/Macht gegenüber der grundlegenden, sozialisierten Kooperation im KDE-Projekt. Damit die Institutionen der Unternehmen, Bezugsgruppe und Ingenieurwissenschaften nicht in einem ständigen Konflikt mit den sozialisierten Institutionen der selbstorganisierten Gruppen, außeruniversitären Professionen, Wissenschaft und sozialen Open Source-Bewegungen stehen, wird punktuell an passenden, sozialisierten Institutionen angeknüpft.

Anknüpfungspunkte bestehen z.B. in der Wissenschaft. Erleichtert wurde die Übernahme der hierarchischen Elemente von Unternehmen und Ingenieurwissenschaften, da Release bzw. Endtest eine starke Ähnlichkeit mit Prüfungen oder Abgabeterminen an Hochschulen bzw. mit Publikationsterminen bei wissenschaftlichen Zeitschriften oder bei der Beantragung von Forschungsprojekten als hierarchische Elemente haben. Diese Prüfungen, Abgabetermine oder Publikationstermine wurden dabei während der Hochschulphase von den KDE-Mitgliedern sozialisiert. Die Professionsmitglieder, wissenschaftlichen Gutachter/Zeitschriftenherausgeber bzw. an den Hochschulen die Professoren kontrollieren dann die erbrachte Leistung ihrer Kollegen. Diese Posten des Gutachters/Zeitschriftenherausgebers oder Vorgesetzten, wie des Releasekoordinators beim KDE-Projekt, haben dabei den bedeutendsten Einfluss durch ihre Torwächterfunktion bzw. Verhinderungsmacht. Eine besondere Ähnlichkeit besteht zwischen Zeitschriftenherausgeber und Releasekoordinator, da der Herausgeber letztendlich über das spätere Erscheinen bestimmt, aber auf die Expertise der Gutachter bzw. beim KDE-Projekt des Kreises von Unterstützern mit hoher Reputation vertraut. Die Gutachter kooperieren dabei nicht nur mit dem Herausgeber, sondern auch mit dem Autor, indem sie ihm ein Feedback geben, auch wenn sie ein Skript ablehnen (vgl. Bühl 1974: 87f.; Knorr-Cetina 2002; Heintz 2000a: 181ff.; Merton 1985: 172ff.; Freidson 2001: 72ff., 86, 94). Auch bei der sozialen Bewegung koordiniert oftmals eine zentrale Gruppe mit einer bestimmenden Person einen Protesttermin, wobei auch Macht über die Gestaltung wie im Kde-Projekt besteht (vgl. Oliver, Marwell 1992: 265ff.).

- Eine nachgeordnete *Kooperation* als Grundkonsens bestimmt die Arbeitszusammenführung. Die sozialisierten Institutionen der selbstorganisierten Gruppen, außeruniversitären Professionen, Wissenschaft und sozialen Bewegungen sind weiterhin bestimmend. Eine Kooperation liegt in der Wissenschaft bei einer Publikationserstellung mit mehreren (Ko-)Autoren ständig, auch kurz vor der Abgabe, vor, wobei wie im KDE-Projekt vor allem auf die Fehlerfreiheit der Publikation geachtet wird (s.o.). Eine Kooperation während eines Releases kommt eher der Mobilisierung einer sozialen Bewegung vor konkreten Protesten wie Demonstrationen nahe. Die Protestvorbereitung erfolgt modular, indem die Leiter von verschiedenen Gruppen auf eine Mitwirkung angesprochen werden und diese wiederum ihre Gruppen mobilisieren, die die detaillierte Vorbereitung durchführen (vgl. Oliver, Marwell 1992: 265ff.).

- Der unbedeutende *Wettbewerb* basiert auch auf Institutionen der selbstorganisierten Gruppen, außeruniversitären Professionen, Wissenschaft und sozialen Bewegungen. Wettbewerb erfolgt ähnlich zum KDE-Projekt bei Publikationsabgabe um das Erreichen der wissenschaftlichen Standards, Reputation des Verfassers, aber auch um den Neuigkeitswert. Der Neuigkeitswert spielt als Auswahlkriterium bei traditionellen, aber auch elektronischen wissenschaftlichen Zeitschriften mit standardisierter Seitenzahl eine große,

---

<sup>266</sup> s. z.B. <http://developer.kde.org/development-versions/kde-3.0-release-plan.html>

im Internet wegen des unlimitierten Platzes eine geringere Rolle, was sich auch beim KDE-Projekt zeigt (vgl. Bühl 1974: 87f.; Hinner 2003:129ff.; 188ff.).

### **6.2.3. Zusammenfassung: Analyse der Koordinationsformen anhand des Neo-Institutionalismus**

In diesem Kapitel wurde die Ausprägung und die Bedeutung der Koordinationsformen im Open Source-Projekt KDE anhand einer Theorieinterpretation des Neo-Institutionalismus analysiert. Die Analyse erfolgte wie bei der Neuen Institutionenökonomie in zwei Stufen, die sich jeweils in zwei kleinere Zwischenschritte unterteilen ließen.

Die erste Stufe der Analyse bestand aus einer Theorieinterpretation des Neo-Institutionalismus, die die Theorie im Hinblick auf eine Theorie der Koordination in Organisationen modifiziert. Institutionen werden über verschieden stark institutionalisierende Strukturangleichungsmechanismen von Akteuren aus der gesellschaftlichen Umwelt in das untersuchte Open Source-Projekt KDE übertragen, um legitim gegenüber der gesellschaftlichen Umwelt zu erscheinen. Strukturangleichungsmechanismen sind neben (juristischen oder informalen) Sanktionen und der Imitation von vorbildhaften Problemlösungsmustern das normorientierte Verhalten durch Sozialisation. Das KDE-Projekt benötigt Legitimität, um Ressourcen von den Akteuren aus der gesellschaftlichen Umwelt zu erhalten. Die gesellschaftliche Umwelt besteht aus einer Bezugsgruppe des KDE-Projekts mit anderen Open Source-Projekten und Akteuren aus Parsons Gesellschaftsbereichen bzw. Teilsystemen wie Gemeinschafts-, Kultur-, Politik- und Wirtschaftssystem.

Da das KDE-Projekt in soziale Beziehungen eingebettet ist, bestehen ähnliche Institutionen, hier Koordinationsformen, in der gesellschaftlichen Umwelt, die als Koordinationsformen im KDE-Projekt unterschiedlich stark institutionalisiert werden. Der Strukturangleichungsmechanismus Sozialisation institutionalisiert am stärksten, dann folgt die Sanktion und am geringsten die Imitation. Weiterhin richtet sich das KDE-Projekt an Ressourcen aus, die aber eher im Hintergrund stehen. Die Stärke der Institutionalisierung von Koordinationsformen der Akteure im KDE-Projekt ist auch abhängig von der Wichtigkeit der Ressourcen. Die wichtigste Ressource sind (potenzielle) Mitarbeiter als Personal bzw. Quellcode als Arbeitszeit des Personals, dann folgen materielle, finanzielle Spenden zweitwichtigste und Anerkennung als drittwichtigste Ressource.

Die IuK-technische Infrastruktur kann zwar die Strukturangleichungsmechanismen unterstützen, dies hat aber keine Auswirkungen auf die Institutionalisierung. Eine Sozialisation erfolgt in Gruppen unabhängig von spezifischer Technik, oft aber durch persönliche Treffen, wie oben diskutiert wurde. Die Sanktionen kommen bisher weitestgehend auch ohne IuK-Technik aus, was auch für Imitation gilt. Bei Imitation können Informationen zu Best Practice-Modellen zwar schneller über IuK-Technik verbreitet werden, es ist jedoch ein Medium unter vielen und nicht unbedingt vertrauenswürdig (s. Kap. 6.1.).

Die Trennung von Sozialisation und Profession und die analytische Trennung von Strukturangleichungsmechanismen und Ressourcen war, wie gezeigt, für das Open Source-Projekt anwendbar. Durch diese analytische Trennung konnte aufgezeigt werden, dass weitestgehend jeweils ein Strukturangleichungsmechanismus zusammen mit einer Ressource auftrat. Sozialisation war wie beschrieben vor allem mit der Ressource Personal, Sanktionen mit materiellen/finanziellen Spenden und Imitation mit Anerkennung verbunden.<sup>267</sup> Zudem ist die Betrachtung der gesellschaftlichen Umwelt des Open Source-

---

<sup>267</sup> Die Zusammenhänge zwischen Mechanismen und Ressourcen stellen sich folgendermaßen dar: Projekteintretendes Personal hat bekanntermaßen die Institutionen schon sozialisiert. Sanktionen wirken am ehesten über den Entzug von materiellen/finanziellen Ressourcen. Imitiert werden Verfahren von Or-

Projekts mittels Parsons Gesellschaftstheorie und der Bezugsgruppe für das Open Source-Projekt KDE geeignet, da damit relevante Ausprägungen der relevanten gesellschaftlichen Akteure abgebildet werden konnten.<sup>268</sup>

Obwohl die Koordinationsformen in der Literatur manchmal nicht genau dargestellt werden und eine vollständige Nennung nicht sicher ist, war dieses Vorgehen für das Aufzeigen von Ähnlichkeiten zwischen Koordinationsformen in KDE-Projekt und Akteuren der Gesellschaftsbereiche gut zu verwenden. Es konnten größtenteils die Koordinationsformenarten wie Verhandlungen mit Konsens, Machtausübung u.a. von Posten etc., nicht dagegen Elemente wie Vertrauen, Angst vor Strafe als potenzielle Machtausübung in den gesellschaftlichen Akteuren nachgewiesen werden, die über die Strukturangleichungsmechanismen im KDE-Projekt übernommen wurden.

Anhand des modifizierten Neo-Institutionalismus konnte auch die Bedeutung der Koordinationsformen im KDE-Projekt verglichen mit der Bedeutung ähnlicher Koordinationsformen in den Akteuren der Gesellschaftsbereiche bestimmt werden. Die Bedeutung der Koordinationsformen in den wichtigsten gesellschaftlichen Akteuren spiegelte sich damit auf einer allgemeinen, projektumfassenden Ebene im KDE-Projekt wider.

Dagegen werden die Koordinationsformen in den Teilprozessen eher grob analysiert, da in der empirischen Literatur über den jeweiligen Akteur i.d.R. kein unterschiedlicher Stellenwert der Koordinationsformen für einen Teilprozess dargestellt wird. Trotz dieser Einschränkung konnten durch Assoziationen zwischen KDE-Projekt und den wichtigsten Akteuren bzgl. Teilprozessen einige Vermutungen gewonnen werden. Als möglicher Ansatzpunkt für weitere empirische Forschung wäre zu analysieren, welchen Stellenwert eine Institution im Teilprozess eines Akteurs hat und wie Institution und ihr Stellenwert im Teilprozess z.B. des KDE-Projekts institutionalisiert wird.

Weiterhin konnte festgestellt werden, dass bestimmte Institutionen zusammen auftreten. Beispielsweise sind bei Torwächterposten als hierarchisches Element auch wettbewerbliche Institutionen vorhanden. Für eine Vereinfachung einer Analyse wären solche zusammen auftretenden Institutionen theoretisch und empirisch zu eruieren und theoretisch zu durchdringen. In diesem Zusammenhang kann auch folgende in der Empirie vorgefundene Tendenz erwähnt werden, dass über Sozialisation ein komplettes Institutionengerüst, über Sanktionen ein Ausschnitt aus einem Institutionengerüst und über Imitation nur kleine Änderungen im Institutionengerüst in einem Untersuchungsobjekt bestimmt werden. Diese Tendenz wäre ebenso theoretisch und empirisch aufzuarbeiten.

In einem ersten Zwischenschritt der zweiten Stufe wurde analysiert, wie die Koordinationsformen als Institutionen der Akteure der Gesellschaftsbereiche/-systeme nach Parsons das KDE-Projekt beeinflussen. Dabei wurden Vermutungen entwickelt, aus welchen Akteuren der Gesellschaftsbereiche die Koordinationsformen im KDE-Projekt wie stark verkehrt werden. Zentrale Koordinationsformenkombinationen als Institutionen gehen von relevanten, zentralen Akteuren wie selbstorganisierten Gruppen im Gemeinschaftssystem, außeruniversitären Professionen und den grundlagenwissenschaftlichen Disziplinen Ma-

---

ganisationen mit hoher Anerkennung. Zusätzlich ergeben sich Fragen, ob dies nur für das Open Source-Projekt oder auch für andere Organisationsformen gilt und ob andere Beziehungen zwischen Strukturangleichungsmechanismen und Ressource für bestimmte Organisationsformen bestehen.

<sup>268</sup> Die Begrifflichkeit der technischen und institutionellen Umwelten sowie des organisationalen Feldes und gesellschaftlichen Sektors kann dagegen nicht differenziert genug Einflüsse aus der gesellschaftlichen Umwelt des Open Source-Projekts darstellen bzw. bestimmte Einflüsse überbetonen oder übersehen.

thematik und Physik im Kultursystem sowie der sozialen Open Source-Bewegung im Politiksystem aus. Die Koordinationsformenkombinationen dieser Akteure sind zentral, da diese über die am stärksten institutionalisierende Sozialisation und wichtigste Ressource Personal direkt in die Aktivitätsstruktur des KDE-Projekts übernommen werden. Alle drei Koordinationsformen sind in und bei diesen Akteuren wie im KDE-Projekt nachweisbar. Aus der verwendeten Literatur kann entnommen werden, dass bei den relevanten Akteuren eine (gruppenorientierte) Kooperation als Grundlage der Zusammenarbeit dominiert, wobei eine nachgeordnete Reputationshierarchie mit Posten eine Rolle spielt. Ein Wettbewerb um Reputation besteht verdeckt, ein weiterer Wettbewerb wird eher vermieden, da sonst Konflikte entstehen würden.

Zudem wird Legitimität gegenüber dem Staat/der staatlichen Verwaltung und dem juristischen System benötigt, um Markenrechte, Softwarelizenzrechte (GPL) etc. einfordern und juristisch sanktionieren zu können. Da zusätzlich materielle und finanzielle Spenden mit Sanktionsmöglichkeiten an das KDE-Projekt gehen, wurde ein Verein als Formalstruktur, entkoppelt von der Aktivitätsstruktur, anhand juristischer Vorschriften im KDE-Projekt gegründet, um Legitimität gegenüber Spendern, Staat und juristischem System aufzubauen und die Spenden kanalisieren zu können. Staat/staatliche Verwaltung und juristisches System beeinflussen das KDE-Projekt weder über Spenden noch direkte Kontakte.

Die Koordinationsformen in der Aktivitätsstruktur der ablaufenden Softwareentwicklung des KDE-Projekts stammen nicht von den Institutionen des Vereins, da eher Hochschulen mit Bezug zur Wissenschaft neben Unternehmen sporadisch durch Spenden das KDE-Projekt unterstützen. Außerdem bestehen eher direkte Kontakte zu Open Source-bewegungsnahen Unternehmen als zu anderen Unternehmen. Dieser geringe Kontakt zu Unternehmen und Unternehmensangehörigen und die Verwendung von Spenden statt Verträgen kann auch erklären, warum im KDE-Projekt eine strategische Kooperation keine Rolle spielt. Dagegen haben beim Gnome- und Debian-Projekt die Elemente eines Vereins vermutlich einen sehr starken Einfluss durch einen starken Kontakt und Arbeitseinsatz von Unternehmensangehörigen oder einer höheren Spendenabhängigkeit von Unternehmen auf deren Aktivitätsstruktur. Dies zeigt auch minimale Unterschiede zwischen den sonst stark ähnelnden Open Source-Projekten der Bezugsgruppe auf, wie sie auch in der Literatur zum Neo-Institutionalismus u.a. auch gesehen wurden.

Auch die anderen Open Source-Projekte aus der Bezugsgruppe des KDE-Projekts beeinflussen die Koordinationsformen in der Aktivitätsstruktur kaum. Das KDE-Projekt imitiert nur technische Lösungen wie Quellcode, fertige Programme oder technische Ideen aus seiner Bezugsgruppe. Die Imitation als institutionalisierungsschwächster Strukturangleichungsmechanismus, gepaart mit der Anerkennung von der Bezugsgruppe als dritt wichtigste Ressource lässt das KDE-Projekt keine weiteren Institutionen importieren.

In einem zweiten Zwischenschritt wurde grob analysiert (s.o.), welche Koordinationsformen als Institutionen der Akteure in den Teilprozessen der Softwareentwicklung wirken. Die Bedeutung der Koordinationsformen konnte über die Bedeutung der Institutionen der Akteure festgestellt werden, die oftmals einem Teilprozess im KDE-Projekt zugeordnet werden konnten. Es wurden vor allem die Institutionen der Akteure einbezogen, die über Sozialisation und die Ressource Personal besonders stark im KDE-Projekt institutionalisiert wurden. Folglich war eine Koordinationsform in einem Teilprozess im KDE-Projekt besonders bedeutend, die von vielen ähnlichen Koordinationsformen der Akteure der Gesellschaftsbereiche institutionalisiert wurde. Zugleich konnte eine hohe Bedeutung von bestimmten Koordinationsformen der zentralen Akteure wie einer Basiskooperation vorgefunden werden, die in mehreren Teilprozessen einen Einfluss hat.

Generell werden die Teilprozesse Eintritt/Rekrutierung, Arbeitsplanung (Zieldefinition, Arbeitsverteilung von Arbeitspaketen mit detaillierter Entwicklungsplanung, Arbeitsverteilung von Posten), Leistungserstellung mit detaillierter Entwicklungsplanung und Arbeitszusammenführung durch die Koordinationsformenkombinationen von zentralen Akteuren wie selbstorganisierten Gruppen, außeruniversitären Professionen, grundlagenwissenschaftlichen Disziplinen Mathematik sowie Physik und sozialen Bewegungen bestimmt. In diesen Teilprozessen konnten mehr kooperative Elemente aus den Akteuren nachgewiesen werden als hierarchische. Zudem hatte auch in diesen Teilprozessen tendenziell die Hierarchie/Macht in den relevanten Akteuren einen geringeren Stellenwert als die Kooperation. Wettbewerb spielte eine ähnlich verdeckte Rolle bzw. wird in den Akteuren vermieden wie im KDE-Projekt.

Bei Eintritt/Rekrutierung und bei Arbeitsverteilung durch Postenbesetzung bei der Arbeitsplanung wirken aber eher die Koordinationsformenkombinationen der selbstorganisierten Gruppe bzw. der sozialen Bewegung als der außeruniversitären Profession und wissenschaftlichen Disziplinen. Beim Eintritt ist Wettbewerb nach Hierarchie/Macht die zweitwichtigste Koordinationsform, da bei der selbstorganisierten Gruppe bzw. der sozialen Bewegung beim Eintritt neuer Mitglieder auch eine wettbewerbliche Auswahl durch den Posteninhaber als dominierendes hierarchisches Element erfolgt. Bei der Postenübernahme durch KDE-Mitarbeiter wird wie in selbstorganisierten Gruppen gehandelt. Eine nachgeordnete Kooperation als Grundkonsens wurde ins KDE-Projekt auch von den anderen beiden Akteuren importiert. Ein Wettbewerb besteht nicht, da in der dominierenden selbstorganisierten Gruppe kein Wettbewerb bei der Postenbesetzung vorhanden ist.

Bei der Arbeitszusammenführung ist Hierarchie/Macht am bedeutendsten, da für eine Arbeitszusammenführung neben den sozialisierten Institutionen der selbstorganisierten Gruppen, außeruniversitären Professionen, Wissenschaft generell und sozialen Bewegungen auch einige imitierte und sanktionierte neben sozialisierten Institutionen von Unternehmen und den Ingenieurwissenschaften eine Rolle spielen. Da aber die sozialisierten Institutionen der selbstorganisierten Gruppen, außeruniversitären Professionen, Wissenschaft und sozialen Bewegungen weiterhin bestimmend sind, ist Kooperation zwar nachgeordnet, jedoch immer noch bedeutend. Ein Wettbewerb um den Neuigkeitswert neuer Beiträge ist wie in der Wissenschaft durch den Einfluss des Internets gering.

## 7. Zusammenfassung und Ausblick

### 7.1. Koordination der Softwareentwicklung in einem Open Source-Projekt

Ausgangspunkt dieser interdisziplinären Arbeit war die Feststellung, dass neue Arbeitsformen wie virtuelle Arbeitsnetze, darunter auch Open Source-Projekte, im Rahmen einer Differenzierung und Diversifizierung der gesellschaftlichen Arbeit entstehen. Aufgrund der vernetzten und vernetzenden IuK-Technik nimmt die Bedeutung von Selbständigenarbeit zu und neue Arbeitsformen im Zusammenhang mit Eigen- und Ehrenamtsarbeit entstehen. In diesem Spannungsfeld sind virtuelle Arbeitsnetze zu finden. Damit wird in dieser Arbeit eine Thematik aufgegriffen, die bisher kaum in der Literatur über IuK-technisch-vermittelte Arbeit untersucht wurde, die sich eher auf virtuelle standortverteilte (Projekt-)Gruppen in Unternehmen konzentriert.

Im Zentrum dieser Analyse standen also hoch virtuelle Arbeitsnetze, in denen Software informell, temporär von individuellen, rechtlich autonomen Akteuren entwickelt wird.

Ein Beispiel eines hoch virtuellen Arbeitsnetzes, das Software erstellt, wurde identifiziert, ausgewählt und empirisch untersucht: Den Schwerpunkt dieser Arbeit bildete das Open Source-Projekt KDE, in dem freiwillig, in Form von Eigen- und Ehrenamtsarbeit gearbeitet wird. Im Rahmen dieses Projekts entsteht eine grafische Benutzeroberfläche vor allem für das Open Source-Betriebssystem Linux.

Für diese Untersuchung wurden aus den virtuellen Arbeitsnetzen ein Open Source-Projekt als neu entstandene Arbeitsformen ausgewählt, da dieses von Erwerbsarbeit am weitesten entfernt ist und bisher diese neue Art der Eigen-/Ehrenamtsarbeit nicht aus einer Arbeitsperspektive untersucht wurde. Gerade die Freiwilligkeit gegenüber der Erwerbsarbeit machte den Arbeitsprozess und das Arbeitshandeln im Projekt besonders interessant.

Virtuelle Arbeitsnetze und das untersuchte Open Source-Projekt wurde bisher von der wissenschaftlichen Forschung erheblich vernachlässigt, da hauptsächlich der Einfluss der IuK-Technik auf die Erwerbsarbeit analysiert wurde. Darüber hinaus stehen in Studien über Softwareentwicklung vorzugsweise Unternehmen sowie Phasenmodelle, Arbeitsstile und Projekttypen im Mittelpunkt. Die Regeln des Arbeitsprozesses bei hoher Virtualität in einer neuen temporär-informalen Arbeitsform wurden bisher kaum erforscht, weswegen in dieser Arbeit über diese Ansätze hinaus der Arbeitsprozess und seine Normen untersucht wurde. Damit erfolgte eine erste Auseinandersetzung mit einem Arbeitsprozess im Zusammenhang mit praktischer Kooperation, Netzwerkformen, Nutzung von IuK-Techniken, Wissenstransfer und Arbeit. Darauf aufbauend können weitere Schritte im Hinblick auf einen wie in der Einleitung geforderten, empirisch reichen qualitativen Begriff einer informatisierten Arbeit gemacht werden (s.a. Schmiede 2006: 468).

Zur Untersuchung der Normen im Arbeitsprozess wurden aus Arbeitsmarktfunktionen und Phasenmodellen der Softwareentwicklung Teilprozesse entwickelt, die nicht zeitliche, sondern logische Abfolgen darstellen. Ausgehend von den bisher nur zeitlich abfolgenden Phasen in Vorgehensmodellen geht der neue analytische Ansatz der logisch-abfolgenden Teilprozesse einen neuen Weg, auch um die parallele Vorgehensweise im Open Source-Projekt KDE fassen und untersuchen zu können. Teilprozesse der Softwareentwicklung sind „Eintritt/Rekrutierung“, „Arbeitsplanung mit Zieldefinition, detaillierter Entwicklungsplanung und Arbeitsverteilung“, „Leistungserstellung“ sowie „Arbeitszusammenführung“. Im Teilprozess Arbeitsverteilung werden zum einen Arbeitspakete/-bereiche bei der Softwareentwicklung und zum anderen Posten untereinander aufgeteilt, die entwicklungsunterstützende Aufgaben außerhalb der Softwareentwicklung umfassen. Diese Teilprozesse

stellten den Analyserahmen für die Untersuchung der Normen im KDE-Projekt dar. Dieser Analyserahmen aus Teilprozessen ist ein zusätzlicher Beitrag zur Untersuchung von Netzwerken und eine Weiterentwicklung der Vorgehensmodelle in der Informatik. Er ist ein Ansatzpunkt für wissenschaftliche Vergleiche mit anderen Arbeits- bzw. Softwareentwicklungsabläufen z.B. in Unternehmen.

Aufgrund der Betrachtung von virtuellen Arbeitsnetzen wurde der vorhandene, umfangreiche Netzwerkdiskurs aufgearbeitet. Dabei wurde festgestellt, dass mitunter eine Koordination implizit bei den Organisationsformen Netzwerk, Markt und Unternehmen diskutiert wird. Zunächst konnten in der Literatur, später in der Arbeit Kooperation, Hierarchie/Macht und Wettbewerb als Koordinationsformen nicht nur in Unternehmen oder Märkten, sondern auch in Netzwerken als Organisationsformen nachgewiesen werden. Folglich ist zwischen den Koordinationsformen auf der Handlungsebene und den Organisationsformen zu trennen.

In dieser Weiterentwicklung des Netzwerkdiskurses wurde der Fokus auf die Handlungsebene gelegt, da diese mit dem untersuchten Arbeitsprozess und dem Arbeitshandeln zusammenhängt. Damit wurde zwischen der Organisationsform Netzwerk als Rahmen und den untersuchten Koordinationsformen „Kooperation“, „Hierarchie/Macht“<sup>269</sup> und „Wettbewerb“ als Institutionen in dieser Organisationsform Netzwerk analytisch getrennt. Die Definitionen von Netzwerk, Markt und bürokratischer Organisation (Unternehmen, Verbände etc.) beinhalten aber die Arbeitsteilung als wichtiges Merkmal, wobei die Arbeitsverteilung eine Koordination impliziert. Eine Koordinationsform kann daher zwar dominieren, die Koordinationsformen unterscheiden sich jedoch in ihren Ausprägungen in den Organisationsformen „Netzwerk“, „Markt“ und „Unternehmen“ (s. Kap. 3.2.). Damit ist zwischen den Koordinationsformen auf der Handlungsebene und den Organisationsformen auf einer Me-soebene zu unterscheiden. Auf dieser Handlungsebene wurden die Koordinationsformen systematisch anhand des Kategorisierungsschemas „Handlungsbedingungen, Handlung/Mechanismus und Handlungsergebnis“ definiert, was bisher eklektisch und heuristisch erfolgte. Bei dieser systematischen Definition wurde auf Erkenntnisse aus der Ökonomie, Soziologie, Politologie, Sozialpsychologie und Ethnologie zurückgegriffen, in denen Koordinationsformen als soziale Grundkategorien thematisiert wurden. Damit konnten die Koordinationsformen relativ „überschneidungsfrei“ und vor allem ohne Rückgriff auf diffuse Begriffe wie Moral für Kooperation definiert werden. Die Koordinationsformen als Institutionen auf der Handlungsebene wurden für eine Untersuchung des Arbeitsprozesses der Softwareentwicklung herangezogen.

In der bisherigen Literatur existierte keine systematische Analyse der Koordinationsformen im Arbeitsprozess oder mit Ausnahme von Becher et al. (2007) in sozialen Beziehungen anderer Organisationen. Mit den übergreifenden und interdisziplinären Definitionen der Koordinationsformen wurde die Grundlage für die Untersuchung von Arbeitsnetzen oder anderen Organisationsformen wie Unternehmen geschaffen.

Die Koordinationsformen Kooperation, Hierarchie/Macht und Wettbewerb sind also die Institutionen, die in den Teilprozessen der Softwareentwicklung empirisch im Open Source-Projekt KDE als Fallstudie untersucht wurden. Zentral für die explorative Untersuchung waren leitfadengestützte Interviews. Zusätzliches empirisches Material wurde durch eine Dokumentenanalyse von Mailinglisten, der Projekt-Website und Selbstaussagen auf einer

---

<sup>269</sup> Diese Koordinationsform wurde Hierarchie/Macht genannt, da diese Kategorie einerseits die dynamische Seite der Macht, aber auch deren statische Seite in Form von Posten (Hierarchie) auf der Handlungsebene abbildet. Damit konnte in den untersuchten virtuellen Arbeitsnetzen die Macht in einer relativ breiten Form festgestellt werden.

Homepage, protokollierte teilnehmende Konferenzbeobachtungen sowie einer quantitativen Befragung generiert.

Die drei Koordinationsformen konnten dabei nicht nur im Open Source-Projekt KDE, sondern auch von Becher et al. (2007) im elektronischen Arbeitsmarkt nachgewiesen werden, was die Aussagekraft der Koordinationsformen verstärkt. Die vorgenommene Unterscheidung von Koordinations- und Organisationsform und die Konzentration auf die Handlungsebene mit den Koordinationsformen haben sich somit als sinnvoll erwiesen. Die Koordinationsformen können für unterschiedliche Untersuchungsobjekte unterschiedliche Merkmale annehmen und daher auch für andere Arbeitsnetze und andere Organisationsformen wie Unternehmen verwendet werden. Der bisherige wissenschaftliche Fokus beim Diskurs über Netzwerk, Markt und bürokratische Organisationen (Unternehmen) setzt damit an der falschen Ebene an. Mit diesem neuen Ansatzpunkt kann über die Steuerung von Netzwerken, aber u.U. auch anderen Organisationsformen oder von Softwareentwicklungsprozessen, neu nachgedacht werden. Im Diskurs sollte nicht an der Organisationsform, sondern an den zu steuernden sozialen Beziehungen angesetzt werden.

Ausgehend von den genannten Definitionen wurden also die Ausprägungen und der Stellenwert der Koordinationsformen für die Teilprozesse der Softwareentwicklung im KDE-Projekt festgestellt. Treten die verschiedenen Koordinationsinstrumente als Ausprägungen der Koordinationsformen auf, wird deren Stellenwert anhand ihrer Institutionalisierungsstärke (Verbindlichkeit, Dauerhaftigkeit und Maßgeblichkeit) bestimmt. Diese explorative Untersuchung des KDE-Projekts kann neue Erkenntnisse zum interdisziplinären Diskurs um Open Source-Projekte und zum Diskurs um die Softwareentwicklung in der Informatik beisteuern.

Am Open Source-Projekt KDE als Hauptfokus dieser Arbeit sind individuelle Akteure beteiligt, die sich bestimmten Tätigkeitsbereichen zuordnen lassen. Ungefähr 1000-1200 Personen mit Schwerpunkt in Deutschland arbeiten weltweit in diesem KDE-Projekt. Der überwiegende Teil der Projektmitarbeiter ist demografisch homogen, d.h. männlich, zwischen 20 und 30 Jahre alt und hat oder absolviert eine IT-Ausbildung. Für diese Projektmitarbeiter stellt Spaß am Programmieren die zentrale, intrinsische Motivation dar, wobei auch weitere extrinsische Motivationen wie Erringen von sozialer Anerkennung bzw. Reputation auftreten können. Monetäre Anreize spielen kaum eine Rolle. Die KDE-Mitarbeiter arbeiten hauptsächlich in der Softwareentwicklung. Das Zentrum-Peripherie-Prinzip strukturiert nicht nur die Tätigkeitsbereiche, sondern auch die Softwareentwickler nach Reputation, die Subprojekte mit ihren Softwarebereichen nach Wichtigkeit für das Endprodukt/Beteiligtenanzahl und die Posten wie Projektleiterposten innerhalb des Gesamtprojekts nach Verantwortlichkeit. Eine IuK-technische Infrastruktur, auch unterteilt nach Zentrum und Peripherie, unterstützt die Erstellung einer grafischen Benutzeroberfläche (Desktop) und weiterer Anwendungsprogramme.<sup>270</sup> Sie besteht aus Mailinglisten, einem Chatsystem, einer Website/Homepage und einem Dateiablage-/Versionsmanagementsystem, in dem der Quellcode lagert.

Nachfolgend werden die Erkenntnisse zu der Frage nach den Ausprägungen und dem Stellenwert der Koordinationsformen in den Teilprozessen des Open Source-Projekts vorgestellt, auch da später aus diesen Erkenntnissen Vermutungen über ihre Beständigkeit

---

<sup>270</sup> Da das KDE-Projekt sich in den Merkmalen der Akteure und der IuK-technischen Infrastruktur nur wenig von den anderen Open Source-Projekten unterscheidet, ist zu vermuten, dass sich die Koordination im KDE-Projekt und anderen Open Source-Projekten ähnelt.

angestellt werden. Kooperation kann sich bei der Softwareentwicklung innerhalb des Open Source-Projekts KDE gegenüber Hierarchie/Macht stark entfalten, da Hierarchie/Macht hier nur begrenzt wirksam ist. Wettbewerb spielt kaum eine Rolle.

Allgemein hat Hierarchie/Macht im KDE-Projekt eine geringe Ausprägung, weil

- durch die GPL-Lizenz rechtlich kein Projektbeteiligter oder Projektexterner von der Veränderung oder Nutzung abgehalten werden kann und keiner der proprietären Eigentümer des Quellcodes ist. Damit ist auch die Norm einer freien, ungehinderten, transparenten Informationsweitergabe verbunden.
- es im Open Source-Projekt KDE autonome und gleichberechtigte Mitglieder gibt, die wegen der intrinsischen Motivation nicht zur Mitarbeit gezwungen werden können und jederzeit das Gesamtprojekt durch Unterlassen der Mitarbeit verlassen können.
- es keine physische Bestrafung wegen der hohen Virtualität gibt, da die benötigten Ressourcen immateriell sind und auf der IuK-technischen Infrastruktur als existenzielle Grundlage des Projekts basieren (s.u.). Zudem sind Ressourcen, wie bspw. die Arbeitszeit, größtenteils individuell gebunden.
- alle Aktionen von (Sub-)Projektleitern und Softwareentwicklern mit hoher Reputation wegen der Freiwilligkeit argumentativ bestehen müssen, wobei formales Bildungskapital bzw. -zertifikate keine Rolle spielten.

Bei Betrachtung der Koordination in den Teilprozessen dominiert *Kooperation* bei Zieldefinition und Arbeitsverteilung von Arbeitspaketen mit detaillierter Entwicklungsplanung des Teilprozesses Arbeitsplanung sowie beim Teilprozess Leistungserstellung mit detaillierter Entwicklungsplanung. In diesen Teilprozessen tritt besonders die gruppenbezogene Kooperation auf. Sie wird außer bei der Zieldefinition durch eine bedeutende reziproke Kooperation flankiert. Dagegen ist Kooperation bei der Arbeitszusammenführung und Arbeitsverteilung durch Postenbesetzung, die nur auf reziproker Kooperation beruht, der bedeutenderen *Hierarchie/Macht* nachgeordnet. Bei beiden Teilprozessen entscheidet ein Posteninhaber zum einen über die Integration des Quellcodes in den Release, zum anderen über den möglichen Nachfolger. Auch bei Eintritt/Rekrutierung bestimmt ein Projektleiter über die Aufnahme ins KDE-Projekt, wobei eine reziproke Kooperation nachgeordnet auftritt. Posteninhaber oder Personen mit hoher Reputation beeinflussen die restlichen Teilprozesse vor allem über Experten-, Informations- und Identifikationsmacht, da die Übernahme von Aufgaben oder Argumenten nicht erzwungen werden kann. Textbasierte Sanktionen werden manchmal gegen verbale Normübertretungen angewendet. Selten wird fehlerhafter Quellcode gelöscht oder der Schreibzugriff auf das Versionsmanagementsystem entzogen. Das Unterlassen von bestimmten Verhalten kann einerseits durch Angst vor Strafe, aber auch durch die Orientierung an einer Gruppennorm (gruppenorientierte Kooperation) erfolgen. Weitere theoretische Überlegungen über das Nicht-Auftreten von Verhaltensweisen wäre im Rahmen weiterer Forschung zu vertiefen. Ein *Wettbewerb* um Anerkennung/Reputation, Meinungsführerschaft (Konflikte) und Aufmerksamkeit tritt eher verdeckt auf, da Wettbewerb Bestandteil der Softwareentwicklung ist und sich als Motivation darstellt. Die durch die Arbeitsanstrengungen erreichte Reputation kann dann für Hierarchie/Macht verwendet werden.

Im KDE-Projekt dominiert zwar Kooperation, aber in bestimmten Teilprozessen dominiert Hierarchie/Macht. Dieses Ergebnis überrascht, da bisher in der Literatur immer von einer alles überragenden Dominanz einer Koordinationsform auch für jeden Teilprozess ausgegangen wurde. Es bestätigt jedoch den Ansatz, dass am konkreten Objekt anzusetzen und spezifische Arbeitsabläufe, Beziehungen oder Handlungen zu untersuchen sind. Vielmehr kann auch das Open Source-Projekt KDE mit freiwilliger Arbeit ohne Hierarchie/Macht und

einem, wenn auch versteckten Wettbewerb nicht funktionieren. Unterschiedliche Ausprägungen spielen eine Rolle, die in bestimmten Situationen des Arbeitsprozesses offenbar mögliche Nachteile der dominierenden Koordinationsform Kooperation ausgleichen.

Mit dieser Untersuchung ist eine Reflexion über und Einflussnahme auf Einsatz und Veränderung der Steuerungsmodalitäten auch in andere Organisationen möglich. Empirische Vergleiche zwischen Organisationsformen machen die Übernahme von Koordinationsformenelementen von einer Organisationsform in eine andere deutlich, d.h. ob und wie z.B. eine wettbewerbliche Koordinationsform aus dem Markt in ein Unternehmen bzw. Netzwerk übernommen wird. Mit diesem Ansatz wird zudem festgestellt, inwiefern diese übernommene Koordinationsform an die neuen Gegebenheiten an- und in das bestehende Institutionengerüst einer Organisationsform eingepasst wird oder an Grenzen stößt. Zusätzlich können bestimmte Zusammenhänge zwischen Koordinationsformen bestehen, so dass beispielsweise neben einer dominanten Koordinationsform nur bestimmte Koordinationsformen auftreten.

## **7.2. Zusammenhang zwischen luK-Technik und Koordinationsformen sowie zwischen Endnutzern und Softwareentwicklern**

Bei Betrachtung des Zusammenhangs zwischen luK-Technik und Koordinationsformen konnte festgestellt werden, dass die luK-Technik virtuelle Arbeitsnetze und damit Open Source-Projekte erst ermöglicht und durch deren hohe Virtualität bedingt die Koordination bestimmen kann. Von der luK-Technik als virtuelles Spiegelbild der realen Verhältnisse kann nicht für das Open Source-Projekt KDE gesprochen werden, da luK-Technik und Handeln direkt zusammenhängen (vgl. Schmiede 2006). Die auch untersuchte luK-technische Infrastruktur beeinflusst und unterstützt als statisches, gekapseltes Artefakt nicht nur die Softwareentwicklung, sondern auch das Endprodukt. Die Software und damit auch die Infrastruktur wird als Maschinencode genutzt, aber auch im KDE-Projekt nach den eigenen Bedürfnissen geformt, weiterentwickelt und angepasst. Damit besteht eine Wissensspirale, bei der Wissen direkt durch neues Wissen angereichert werden kann. Die Infrastruktur mit Kommunikations-, Informations- und Produktionsausrichtung kann damit direkt an neue Normen angepasst werden. Folglich determiniert die verwendete luK-technische Infrastruktur den Arbeitsprozess mit seinen Normen nicht, kann aber spezifische Normen unterstützen.

Daher war im KDE-Projekt kein determinierender Einfluss der *luK-technischen Infrastrukturtechnik auf die Koordinationsformen* zu erkennen. Es werden weder Kooperation oder Hierarchie noch Wettbewerb durch die luK-technische Infrastruktur gefördert, sondern alle drei Koordinationsformen unterstützt. Dies liegt daran, dass die Infrastruktur mit dem Versionsmanagementsystem für die konkrete Programmierung sowie Email, ggf. Chat und Website für die Kommunikation grundlegend für die Existenz dieses virtuellen Arbeitsnetzes ist. Es fehlen aber spezifische Werkzeuge, die den Wettbewerb unterstützen wie automatisierte Programme für politische Mehrheitswahlen oder Vergleichs-/Matchinginstrumente, weswegen spezifische Wettbewerbselemente nicht vorzufinden sind. Die genutzte luK-technische Infrastruktur hat also einen direkten Einfluss auf die auftretenden Elemente der Koordinationsformen, aber nur einen mittelbaren auf die Bedeutung der Koordinationsformen. Fehlen bestimmte Werkzeuge/Funktionalitäten der Infrastruktur, können Elemente der Koordinationsformen nicht auftreten und die Bedeutung dieser Koordinationsform kann sinken.

Der Einfluss der freiwilligen Arbeit im Open Source-Projekt KDE zeigt sich bei dem Zusammenhang zwischen Infrastruktur und Hierarchie/Macht. Während textbasierte Sanktionen und Informations- und Expertenmacht häufig angewendet und durch die Infrastruktur als Medieninhalte generell unterstützt werden, hängen die gelegentlichen Quellcodelöschungen sowie die seltene Vergabe und der Entzug von Schreibzugriffen mit dem Aufbau der Infrastruktur als Artefakt direkt zusammen. Da Schreib- und Lesezugriffe zur Infrastruktur die Beteiligung an der Softwareentwicklung erst ermöglichen, kann an dieser Stelle Hierarchie/Macht ansetzen. Ihre Verwendung und Bedeutung für diese Koordinationsform ist aber durch andere Institutionen des KDE-Projekts bestimmt. Die unkontrollierte Leseberechtigung wird nicht eingeschränkt, da Freiwilligkeit, die spezifische GPL-Lizenz und die Norm der freien, ungehinderten, transparenten Informationsweitergabe dagegen stehen (s.o.). Quellcodelöschung und Schreibzugriffentzug schließen dabei Mitarbeiter aus der freiwilligen Softwareentwicklung aus.<sup>271</sup>

Beim Zusammenhang zwischen IuK-technischer Infrastruktur und den Teilprozessen kann keine determinierende, aber auch keine eindeutige Beziehung festgestellt werden. Über die hohe Virtualität des KDE-Projekts werden alle Teilprozesse durch die Infrastruktur unterstützt, damit die eher selten im Internet auftretenden freiwilligen Mitarbeiter mit intrinsischer Motivation und Programmierkenntnis, die weltweit über das Internet verbreitet sind, zusammenkommen und arbeiten können. Wegen der Freiwilligkeit sind auch Reisekosten von den insgesamt weniger finanzstarken Mitarbeitern selbst zu tragen, weswegen persönliche Treffen überwiegend im Rahmen einer jährlichen Konferenz stattfinden.

Verschiedene Interessensgruppen wie Endnutzer eines Softwareprodukts können aus der Umwelt eines Arbeitsnetzes Einfluss auf diese nehmen, da die Softwareentwicklung nicht determiniert ist. Im Open Source-Projekt KDE besteht eine *Beziehung zwischen Endnutzern und Softwareentwicklern*. Beim KDE-Projekt sind die hauptsächlich individuellen Endnutzer wegen der freiwilligen Arbeit teilweise in die Softwareentwicklung mit einbezogen. Zudem sind die Softwareentwickler gleichzeitig auch Nutzer der Software. Dabei sind die Endnutzerbezüge bei Arbeitsverteilung und Leistungserstellung mit dem Eintritt/der Rekrutierung verbunden. Aus Sicht der Endnutzer/potenziellen Mitarbeiter überschneiden sich die Teilprozesse, was als eine fließende, halbdurchlässige Projektgrenze in diesen Teilprozessen zu interpretieren ist. Die Endnutzer wirken auf die detaillierte Entwicklungsplanung bei der Leistungserstellung und der Arbeitsverteilung von Arbeitspaketen ein und können bei beständiger Quellcodeerstellung im Teilprozess Eintritt/Rekrutierung in das KDE-Projekt aufgenommen werden. Diese halbdurchlässige Projektgrenze zeigt sich in diesen Teilprozessen durch Experten- und Informationsmacht der Softwareentwickler gegenüber den Endnutzern.

Da die Softwareentwickler freiwillig an den Programmen arbeiten, müssen Beiträge bzw. Quellcode zeit- und themennah zur Softwareentwicklung sowie fehlerfrei sein, um wahrgenommen und übernommen zu werden (vgl. Brand, Holtgrewe 2007). Diese fließende, halbdurchlässige Grenze sorgt also dafür, dass zum einen Ideen von außen zur Verbesserung des Produkts beitragen und dadurch auch potenzielle Softwareentwickler gewonnen werden können. Zum anderen lenken dadurch für das KDE-Projekt und den Arbeitsprozess unwichtige Beiträge nicht von der freiwilligen Arbeit im Projekt ab.

---

<sup>271</sup> Ähnlich ist es beim elektronischen Arbeitsmarkt, in dem die IuK-Technik die Nutzung von Koordinationsformen nicht determiniert, sondern die Koordination wird in die bestehenden Institutionen des Arbeitsmarkts eingepasst (vgl. Becher et al. 2007).

Zusammengefasst haben die Endnutzer wegen der freiwilligen Arbeit eine Einflussmöglichkeit auf die Softwareentwicklung, obwohl Hierarchie/Macht von den KDE-Softwareentwicklern ausgeht.

Generell wird mit dieser explorativen Untersuchung des Zusammenhangs zwischen IuK-technischer Infrastruktur und Koordinationsformen bzw. Teilprozessen sowie zwischen Endnutzern und Softwareentwicklern ein Beitrag zum techniksoziologischen Diskurs über IuK-Technik geleistet. Für einen vertieften Einblick und eine Identifikation weiterer Einflüsse auf die Nutzung der IuK-Technik ist eine ethnologische Untersuchung wie bei den Laborstudien (vgl. Kap. 2.2.) sinnvoll. Eine Beobachtungsstudie der Nutzung der IuK-technischen Infrastruktur oder eine „teilnehmende Beobachtung“ als Mitarbeiter der Techniknutzung vor Ort kombiniert mit einer quantitativ-standardisierten Befragung zeigen dann weitere Zusammenhänge z.B. durch Institutionen auf. In diesem Zusammenhang ist zu untersuchen, wie und zu welchem Zweck die IuK-technischen Werkzeuge in die Organisationsstruktur eingebettet bzw. implementiert werden. Vergleiche zwischen verschiedenen Organisationen, wie zwischen verschiedenen Arbeitsnetztypen oder unterschiedlichen Unternehmen oder Non-Profit-Verbänden, sind angebracht.

### **7.3. Analytik der vorgefundenen Koordinationsformen im Open Source-Projekt KDE**

Nach der empirischen Auseinandersetzung mit dem untersuchten Open Source-Projekt KDE als virtuelles Arbeitsnetz, wurde an dieser Stelle eine ökonomische und soziologische Analyse dieses Projekts durchgeführt. Vermutungen über die Gründe für das Auftreten spezifischer Ausprägungen und für den Stellenwert der Koordinationsformen wurden angestellt. Bei dieser systematischen Analyse zeigte sich, dass die gewählten Theorien für eine Weiterentwicklung zu einer zukünftigen einheitlichen Theorie der Koordination in Organisationen geeignet sind. Die obige Neudefinition der Koordinationsformen stellt eine Grundlage für eine Theorie der Koordination in Organisationen dar und konnte in die modifizierten Theorien eingepasst werden. In diesem Zusammenhang wurde auch ein Beitrag zur Weiterentwicklung der Neuen Institutionenökonomie und des Neo-Institutionalismus geleistet, indem nicht nur die Koordinationsformen, sondern auch das Open Source-Projekt als Non-Profit-Organisation mit in diese Theorien eingepasst wurde. Auch Kooperation mit Vertrauen konnte entgegen vorheriger Annahmen als aufwandsreduzierendes Instrument in der Neuen Institutionenökonomie eingesetzt werden. Zusätzlich wurden Theoriebestandteile zusammengeführt und die Theorien modifiziert, um mögliche Begründungen über die Bedeutung der Koordinationsformen zu finden. In der Neuen Institutionenökonomie wurden ausgehend von der Transaktionskostentheorie Elemente der Prinzipal-Agent-Theorie integriert. Im Neo-Institutionalismus wurde durch die Zusammenführung mit Parsons Gesellschaftstheorie neue Wege gegangen, um gesellschaftliche Akteure in Gesellschaftssystemen identifizieren zu können.

Die modifizierte Neue Institutionenökonomie bezieht sich besonders auf die Bedeutung der Koordinationsformen im KDE-Projekt. Die Bedeutung wird anhand des (geringen) Aufwands für eine effiziente Koordinationsform bestimmt und mit anderen Koordinationsformen verglichen. Diese Vorgehensweise kann nicht unbedingt für das Auftreten spezifischer Ausprägungen verwendet werden. Dagegen kann der modifizierte Neo-Institutionalismus besonders das Auftreten spezifischer Ausprägungen durch die gesellschaftliche Einbindung des Open Source-Projekts KDE aufzeigen. Besonders Institutionen der relevanten Akteure aus bestimmten Gesellschaftsbereichen wurden im Personal des KDE-Projekts sozialisiert, was zu einer Übernahme bestimmter Ausprägungen in das

KDE-Projekt führt. Zusätzlich kann anhand dieser soziologischen Theorie auch die Bedeutung der Koordinationsformen verständlich gemacht werden. Die Herleitung der Vermutungen zur Bedeutung ist jedoch etwas weniger stabil als bei der Neuen Institutionenökonomie, da in der Literatur nicht alle Koordinationsformen der gesellschaftlichen Akteure detailliert genannt werden und eine vollständige Nennung nicht sicher ist. Für eine eher allgemeine Betrachtung war die Literatur aber gut zu verwenden. Die Koordinationsformen werden in den Teilprozessen eher grob analysiert, da in der empirischen Literatur über den jeweiligen Akteur i.d.R. kein unterschiedlicher Stellenwert der Koordinationsformen für einen Teilprozess dargestellt wird. Trotz dieser Einschränkung konnten durch Assoziationen zwischen KDE-Projekt und den wichtigsten Akteuren bzgl. Teilprozessen einige Vermutungen gewonnen werden. Eine empirische Untersuchung eines Untersuchungsobjekts und beeinflussender Akteure ist daher für eine erhöhte Stabilität der Theoriemodifikation notwendig.

Die Neue Institutionenökonomie und der Neo-Institutionalismus sind somit zwei Ansätze, die ihren Schwerpunkt entweder auf die Ausprägung oder die Bedeutung von Institutionen legen, sich somit kaum widersprechen und eher ergänzen. Zusätzlich schließen sich die beiden Theorien, obwohl beide die Bedeutung von Institutionen erklären können, nicht gegenseitig hinsichtlich der Bedeutung von Institutionen aus. Vielmehr kann davon ausgegangen werden, dass Institutionen, die legitim gegenüber ihrer gesellschaftlichen Umwelt und effizient (mit geringem Aufwand) sind, besonders robust gegen Veränderungen und damit besonders stark institutionalisiert sind. Für eine Weiterentwicklung zu einer Theorie der Koordination in Organisationen sollten daher beide Theorien miteinander verbunden werden. Für eine zukünftige Organisationsforschung und -beratung ist für die Fortführung dieser Gedanken besonders interessant, wie und wann in Organisationen Aufwand oder Legitimität bzw. gesellschaftliche Einbettung eine Rolle spielen und daher Veränderungen möglich sind.<sup>272</sup>

Anhand der modifizierten Neuen Institutionenökonomie und des modifizierten Neo-Institutionalismus konnten Vermutungen über die Koordination im Open Source-Projekt erstellt werden. Bei der Analyse mittels der Neuen Institutionenökonomie wurde festgestellt, dass die IuK-technische Infrastruktur im Zusammenhang mit den Koordinationsformen (z.B. Hierarchie/Macht) Transaktionskosten senken kann und damit auch einen Einfluss auf die Bedeutung der Koordinationsform hat. Beim Neo-Institutionalismus dagegen spielt die IuK-technische Infrastruktur keine Rolle, da die Strukturangleichungsmechanismen unabhängig von Technik funktionieren. Eine strategische Kooperation tritt generell nicht auf, da einerseits transaktionskostensenkendes, kalkulatives Vertrauen nicht vorhanden ist, andererseits kaum ein Bezug zu Unternehmen oder Verträgen mit finanzieller Entlohnung besteht.

Die Bedeutung der Koordinationsformen in den oben genannten Teilprozessen im KDE-Projekt wurde bei der modifizierten Neuen Institutionenökonomie wie erläutert untersucht, da diese Theorie nur die Bedeutung der Koordinationsformen betrachten kann. Bei Eintritt/Rekrutierung werden Transaktionskosten auf die potenziellen Mitarbeiter abgewälzt, indem diese in einem Wettbewerb um Aufmerksamkeit und Anschlussfähigkeit der Beiträge und des Quellcodes sorgen müssen. Damit entstehen dem über die Aufnahme entscheidenden (Sub-)Projektleiter kaum Kosten. Eine gering bedeutende Kooperation ermöglicht über ein Vorvertrauen eine Kontaktaufnahme, da sonst zu hohe Verhandlungs-

---

<sup>272</sup> Der Einfluss von gesellschaftlichen Gruppen, betrachtet durch die Linse des soziologischen Neo-Institutionalismus, steht in Kapitel 6.2 im Fokus. Dabei bestehen theoretische Anknüpfungspunkte für die historischen Sozialkonstruktivisten in Nachfolge von Pinch und Bijker (1987).

kosten entstehen würden. Nach der Überwindung dieser Hürde reduziert eine gruppenorientierte Kooperation mit einem Basiskonsens und hohem Vertrauen die Transaktionskosten aller KDE-Mitarbeiter in allen Teilprozessen stark. Hierarchie/Macht senkt zum einen Transaktionskosten durch die (kollektive) Sanktionierung von schädigendem Verhalten, zum anderen beeinflussen gleichzeitig die mächtigeren KDE-Mitarbeiter den Arbeitsprozess und generieren größtenteils Kommunikationskosten. Diese Einflüsse führen in allen Teilprozessen außer bei der Arbeitsverteilung durch Postenbesetzung und der Arbeitszusammenführung bei Kooperation zu einer hohen, bei Hierarchie/Macht zu einer geringen Bedeutung. Hierarchie/Macht ist in diesen Ausnahmen dagegen bedeutender als Kooperation, da ein Posteninhaber schädigendes Verhalten, mangelnde Fehlerfreiheit und mangelnde Termineinhaltung durch potenzielle Machtausübung/Letztentscheidungsmacht vermeiden kann. Kooperation ist wegen des bestehenden Grundkonsenses mit hohem Vertrauen etwas weniger bedeutend, jedoch würden Verhandlungen mit Konsens zu sehr hohen Transaktionskosten führen, obwohl eine Win-Win-Situation besteht. Wettbewerb spielt in den Teilprozessen generell eine geringe Rolle im Projekt, da redundante Arbeiten und die redundante Ausrichtung der KDE-Mitarbeiter auf einen Wettbewerb zu hohen Transaktionskosten führen würden. Durch diesen Vergleich effizienter Koordinationsformen wurde in dieser Arbeit die Grundlage für weiterführende Analysen über Nachteile der Koordinationsformen gelegt.

Anhand des modifizierten Neo-Institutionalismus konnten folgende relevante, zentrale Akteure aus der Umwelt des KDE-Projekts identifiziert werden, aus denen die Ausprägungen der Koordinationsformen über spezifisch sozialisiertes Personal in das KDE-Projekt institutionalisiert wurden: Akteure sind selbstorganisierte Gruppen aus dem Gemeinschaftssystem, außerwissenschaftliche Professionen und die grundlagenwissenschaftlichen Disziplinen Mathematik und Physik aus dem Kultursystem sowie die soziale Open Source-Bewegung aus dem Politiksystem, wobei auch diese Bewegung einer selbstorganisierten Gruppe ähnelt. Aus der verwendeten Literatur kann entnommen werden, dass bei den relevanten Akteuren eine (gruppenorientierte) Kooperation als Grundlage der Zusammenarbeit dominiert, wobei eine nachgeordnete Reputationshierarchie mit Posten eine Rolle spielt. Ein Wettbewerb um Reputation besteht verdeckt, ein weiterer Wettbewerb wird eher vermieden, da sonst Konflikte entstehen würden. Institutionen aus einer formalen Vereinsstruktur werden nicht in die Aktivitätsstruktur übernommen, da die materiellen und finanziellen Spenden über den möglichen Entzug als Sanktion eher von Universitäten, weniger von Unternehmen und nicht vom Staat/der staatlichen Verwaltung kommen, die die Vereinsstruktur vorschreiben und sanktionieren. Zudem arbeiten vor allem Studenten im KDE-Projekt mit, die diese formalen Vereins-Strukturen kaum kennen. Der gegründete KDE-Verein wird nur zur Kanalisierung von Spenden und zur möglichen legitimen Sanktionierung z.B. der Softwarelizenz über das juristische System herangezogen. Außerdem sind die Vereinsstrukturen größtenteils nicht in den über Sozialisation des Projekt-Personals stark institutionalisierten Koordinationsformen obiger Akteure wiederzufinden. Einen geringen Einfluss auf das KDE-Projekt haben Unternehmen zum einen über sporadische Spenden, zum anderen über gelegentlichen direkten Kontakt zum Projekt, wobei diese Unternehmen der Open Source-Bewegung bzw. dem Projekt zugeneigt sind. Minimale Unterschiede im Arbeitsprozess zwischen den sonst stark ähnelnden Open Source-Projekten der Bezugsgruppe können durch diese Theoriemodifikation anhand des Einflusses des Vereins aufgezeigt werden. Im Gnome- und Debian-Projekt haben die Unternehmen einen größeren Einfluss überwiegend durch angestellte Open Source-Entwickler sowie überlebenswichtige Spenden, weswegen einige formale Vereinsstrukturen wie Wahlen nach dem Mehrheitsprinzip übernommen wurden. Andere Open Source-Projekte als Be-

zugsgruppe werden im Zusammenhang mit Anerkennung imitiert, wenn diese technische Neuerungen hervorbringen.

Bei der Betrachtung der Bedeutung der Koordinationsformen in den Teilprozessen sind folgende Tendenzen zu erkennen: Eintritt/Rekrutierung wird durch die dominierende Hierarchie/Macht der Posteninhaber als Zugangskontrolle und dem damit zusammenhängenden Wettbewerb aus selbstorganisierten Gruppen und der sozialen Open Source-Bewegung bestimmt. Kooperation wurde in diesem Zusammenhang kaum erwähnt. Der Teilprozess Arbeitsverteilung durch Postenbesetzung wird durch dieselben Akteure wie bei Eintritt/Rekrutierung beeinflusst, weswegen Hierarchie/Macht von ähnlich hoher Bedeutung ist. Kooperation ist von mittlerer Bedeutung, da nur die Institutionen von selbstorganisierten Gruppen integriert werden und ein Grundkonsens besteht. Dagegen besteht ein Grundkonsens, der die restlichen Teilprozesse im KDE-Projekt dominiert, auch in selbstorganisierten Gruppen, außerwissenschaftlichen Professionen und den grundlagenwissenschaftlichen Disziplinen Mathematik und Physik sowie der sozialen Open Source-Bewegung. Die nachgeordnete Hierarchie/Macht wurde von weniger Akteuren ins KDE-Projekt übernommen. Zudem hatte auch in diesen Teilprozessen tendenziell die Hierarchie/Macht in den Akteuren einen geringeren Stellenwert als die Kooperation. In der Arbeitszusammenführung wurde nicht nur der Grundkonsens obiger Akteure ins Institutionengerüst des KDE-Projekts eingepasst, sondern neben den üblichen sozialisierten Hierarchie/Macht-Elementen auch hierarchische Posten von Unternehmen, Ingenieurwissenschaften und von der Bezugsgruppe imitiert und über den möglichen Entzug von Spenden sanktioniert. Unterstützend wirken Torwächterposten in Wissenschaft und Profession. Auch Wettbewerb hängt mit bestimmten Torwächterposten in Wissenschaft und Profession zusammen, so dass bestimmte Institutionengeflechte von Koordinationsformen bestehen können. Doch erfolgt der sozialisierte Wettbewerb generell in der Wissenschaft und Profession, aber auch selbstorganisierte Gruppe und soziale Bewegung wie im KDE-Projekt eher verdeckt.

Aus der Neuen Institutionenökonomie und dem Neo-Institutionalismus können also Vermutungen zu einem Open Source-Projekt als Non-Profit-Organisation entwickelt werden. Aus diesen Vermutungen lassen sich Hypothesen generieren und auf alle Open Source-Projekte übertragen. Zusammen mit den Theoriemodifikationen sind die Vermutungen ein Schritt in Richtung eines Vergleichs einer Softwareentwicklung zwischen Open Source-Projekt und anderen virtuellen Arbeitsnetzen oder zwischen Projekt und Unternehmen. Ein Vergleich verschiedener empirisch geprüfter Vermutungen zu den Ausprägungen und Bedeutungen der Koordinationsformen in verschiedenen Organisationen ist notwendig, um eine Theorie der Koordination in Organisationen aufzustellen. Ausgehend von dieser Arbeit mit ihren theoretischen Vorarbeiten kann auch ein allgemeiner Effizienzvergleich der Organisationen ausgehend von einer ähnlichen Software angestrebt werden.

## 8. Literatur:

- (N)onliner Atlas (2004): *Eine Topographie des digitalen Grabens durch Deutschland*. [http://www.nonliner-atlas.de/pdf/NONLINER-Atlas2004\\_TNS\\_Ernid\\_InitiativeD21.pdf](http://www.nonliner-atlas.de/pdf/NONLINER-Atlas2004_TNS_Ernid_InitiativeD21.pdf), 20.04.2007.
- Abbott, A. (1988): *The System of Professions. An Essay on the Division of Expert Labor*. Chicago: Chicago U.P.
- Abrahamsson, P.; Salo, O.; Ronkainen, J.; Warsta, J. (2002): *Agile Software development methods. Review and Analysis*. Espoo, <http://www.vtt.fi/inf/pdf/publications/2002/P478.pdf>, 20.04.2007.
- Achtenhagen L.; Müller-Lietzkow, J.; Knyphausen-Aufseß, D. z. (2003): *Das Open Source-Dilemma: Open Source-Software zwischen freier Verfügbarkeit und Kommerzialisierung*. In: ZFBF, Schmalenbachs Zeitschrift für Betriebswirtschaft, 55 (August 2003), 455-481.
- Adler, P. (2001): *Market, Hierarchy, and Trust: The knowledge economy and the future of capitalism*. In: Organization Science, Volume: 12, Issue: 2, Mar. - Apr. 2001.
- Adloff, F.; Sigmund, S. (2005): *Die gift economy moderner Gesellschaften. Zur Soziologie der Philanthropie*. In: Adloff, F.; Steffen M. (Hrsg.): *Vom Geben und Nehmen. Zur Soziologie der Reziprozität*. Frankfurt/Main: Campus Verlag, 211-236.
- Alchian, A.; Demsetz, H. (1972): *Production, Information costs and economic organisation*. In: American economic review, Vol.62, 777-795.
- Altemeier, J. (1999): *Föderale Finanzbeziehungen unter Anpassungsdruck; Verteilungskonflikte in der Verhandlungsdemokratie*. Frankfurt/Main; New York: Campus Verlag.
- Andel, N. (1983): *Finanzwissenschaft*. Tübingen: Mohr.
- Anderson, A.; Beattie, R.; et. al. (1998): *Chrysler Goes to "Extremes". Case study*. In: Distributed Computing, October 1998.
- Apitzsch, B.; Tünte, M.; Urspruch, T. (2005): *Neue Selbstständige in der Personalstruktur virtueller Projekte*. In: Neuendorff, H.; Ott, B. (Hrsg.): *Unternehmensübergreifende Prozesse und ganzheitliche Kompetenzentwicklung. Neue Forschungsergebnisse und visionäre Instrumente zur Unterstützung virtueller Zusammenarbeit*. Frankfurt/Main: Peter Lang: 171-182.
- Arentzen, U.; Lörcher, U. (Hrsg.; 1997): *Gabler-Wirtschafts-Lexikon*. Wiesbaden: Gabler, 14. Aufl.
- Arnold, O.; Faisst, W.; Härtling, M.; Sieber, P. (1995): *Virtuelle Unternehmen als Unternehmenstyp der Zukunft?* In: HMD, 185/1995.
- Atmaca, D. (2002): *Kooperation im Wettbewerb, Kontinuität im Wandel. Identität und Erfolg der produktivgenossenschaftlichen Organisationsform*. Aachen: Shaker Verlag.
- Axelrod, R. (1984): *The evolution of cooperation*. New York: Basic Books.
- Bachmann, R. (2000): *Die Koordination und Steuerung interorganisationaler Netzwerkbeziehungen über Vertrauen und Macht*. In: Sydow, J.; Windeler, A. (Hrsg.): *Steuerung von Netzwerken*, Opladen: Westdeutscher Verlag, 107-125.
- Backhaus, K.; Plinke, W. (1990): *Strategische Allianzen als Antwort auf veränderte Wettbewerbsstrukturen*. In: Zeitschrift für betriebswirtschaftliche Forschung – Sonderheft Strategische Allianzen (1990) 27, 21-33.
- Balzert, H. (1996): *Lehrbuch der Software-Technik – Software-Entwicklung*. Heidelberg: Spektrum Akademischer Verlag.
- Banse, G.; Grunwald, A.; König, W.; Ropohl, G. (2006, Hrsg.): *Erkennen und Gestalten. Eine Theorie der Technikwissenschaften*. Berlin: Edition Sigma.
- Baukowitz, A.; Berker, T.; Boes, A.; Pfeiffer, S.; Schmiede, R.; Will; M. (2006, Hrsg.): *Informatisierung der Arbeit? Gesellschaft im Umbruch*. Berlin: edition sigma.
- Baukowitz, A.; Boes A.; Schwemmler M. (1998): *Veränderungstendenzen der Arbeit im Übergang zur Informationsgesellschaft – Befunde und Defizite der Forschung*. In: Deutscher Bundestag (Hrsg.): *Arbeitswelt in Bewegung: Trends, Herausforderungen, Perspektiven/ Enquete-Kommission „Zukunft der Medien in Wirtschaft und Gesellschaft; Deutschlands Weg in die Informationsgesellschaft“*. Bonn: ZV Zeitungs-Verlag Service.
- Baukowitz, A.; Boes, A. (2002): *Die Zukunft betrieblicher und gewerkschaftlicher Interessenvertretung in modernen Unternehmensstrukturen*. In: Kock, K.; Kurth, M. (Hrsg.): *Arbeiten in der New Economy. Eine Veröffentlichung der Kooperationsstelle Wissenschaft-*

- Arbeitswelt im Landesinstitut Sozialforschungsstelle Dortmund in Zusammenarbeit mit der Heinrich-Böll-Stiftung NRW. Beiträge aus der Forschung, Bd. 128.
- Becher, C.; Brand, A.; Kawai, A.; Schmid, A. (2005): *Elektronische Arbeitsmärkte. Empirie der Struktur und Funktionsweise elektronischer Arbeitsmarktplätze in Teilarbeitsmärkten*. München und Mering: Rainer Hampp Verlag.
- Becher, C.; Brand, A.; Schmid, A. (2007): *Koordinierung elektronischer Arbeitsmärkte – Wettbewerb, Hierarchie und Kooperation auf elektronischen Teilarbeitsmärkten*. München und Mering: Rainer Hampp Verlag.
- Beck, K. (2000): *eXtreme Programming Explained: Embrace Change*. Reading, Mass.: Addison Wesley Longman.
- Beisheim, M.; Zürn, M. (1999): *Transnationale Nicht-Regierungsorganisationen. Eine Antwort auf die Globalisierung?* In: Klein, A.; Legrand, H.-J.; Leif, T. (Hrsg.): *Neue soziale Bewegungen. Impulse, Bilanzen und Perspektiven*. Opladen: Westdeutscher Verlag.
- Benkler, Y. (2000): *From consumers to users: Shifting the deeper structures of regulation toward sustainable commons and user access*. In: *Federal Communications Law Journal*, Vol. 52, No. 3.
- Benkler, Y. (2002): *Coase's Penguin, or Linux and the Nature of the Firm*. In: *Yale Law Journal* 112 oder <http://www.benkler.org/CoasesPenguin.PDF>, 20.04.2007.
- Benner, C. (2002): *Work in the New Economy. Flexible Labor Markets in Silicon Valley*. Oxford: Blackwell Publishing.
- Bentem, N. (2001): *Freiwillige Vereinigungen – Eine vergessene Größe im Dritten Sektor*. In: Strob, B. (Hrsg.): *Vereintes Lernen. Regionale Lernkulturen und Vereinslandschaften in den alten und neuen Bundesländern*. Münster: Waxmann.
- Berger, P.; Luckmann, T. (1980): *Die gesellschaftliche Konstruktion der Wirklichkeit. Eine Theorie der Wissenssoziologie*. Frankfurt/Main: Fischer, engl. Original 1966
- Bergquist, M.; Ljungberg, J. (2001): *The power of gifts: organizing social relationships in open source communities*. In: *Information Systems Journal*, 2001, 11, 305-320.
- Bernart, Y. (1999): *Selbsthilfegruppen als Paradigma des Vergesellschaftungsprozesses*. In: Schäfers, B. (Hrsg.): *Einführung in die Gruppensoziologie*. Wiesbaden: Quelle & Meyer, 3. Aufl., 287-310.
- Beywl, W. (1987): *Alternative Ökonomie. Selbstorganisierte Betriebe im Kontext neuer sozialer Bewegungen*. In: Roth, R.; Rucht, D. (Hrsg.): *Neue soziale Bewegungen in der Bundesrepublik Deutschland*. Frankfurt/Main: Campus Verlag.
- Bezroukov, N. (1999a): *Open Source Software Development as a Special Type of Academic Research (Critique of Vulgar Raymondism)*. In: *First Monday*, Vol. 4, Nr. 10, [http://firstmonday.org/issues/issue4\\_10/bezroukov/](http://firstmonday.org/issues/issue4_10/bezroukov/), 15.04.2007.
- Bezroukov, N. (1999b): *A Second Look at the Cathedral and Bazaar*. In: *First Monday*, Vol. 4, Nr. 12, [http://firstmonday.org/issues/issue4\\_12/bezroukov/](http://firstmonday.org/issues/issue4_12/bezroukov/), 15.04.2007.
- Bierhoff, H. (1993): *Sozialpsychologie. Ein Lehrbuch*. Stuttgart: Kohlhammer.
- Bierhoff, H.; Klein, R. (1992): *Prosoziales Verhalten*. In: Stroebe, W.; Hewstone, M.; Codol, J.-P.; Stephenson, G. (Hrsg.): *Sozialpsychologie. Eine Einführung*. Berlin: Springer. 2. Aufl.
- Bittner, P. (2003): *Informatik (anders) denken ... Über „gute“ Disziplinarität, Kritische Theorie und Informatik*. In: Böhme, G.; Manzei, A. (Hrsg.): *Kritische Theorie der Technik und der Natur*. München: Fink 2003, 162ff.
- Bittner, U.; Hesse, W.; Schnath, J. (1995): *Praxis der Softwareentwicklung: Methoden, Werkzeuge, Projektmanagement. Eine Bestandsaufnahme*. München, Wien : Oldenbourg.
- Blair, H. (2003): *Winning and losing in flexible labour markets: The formation and operation of networks of interdependence in the UK film industry*. In: *Sociology*, Vol. 37, 4, 677-694.
- Blau, P. (1964): *Exchange and power in social life*. New York: EA.
- Bluhm, K.; Schmidt, R.; Teufel, B. (2003): *Kooperationen und Unterstützungsnetzwerke im Internationalisierungsprozess von europäischen kleinen und mittleren Unternehmen – Eine explorative Studie*. In: Hirsch-Kreinsen, H.; Wannöffel, M. (Hrsg.): *Netzwerke kleiner Unternehmen. Praktiken und Besonderheiten internationaler Zusammenarbeit*. Berlin: Ed. Sigma, 89-110.
- Boehm, B. (2002): *Get ready for the agile methods, with care*. In: *Computer*, 35, 1, 64-69

- Boes, A.; Pfeiffer, S.; Schmiede, R. (2006): *Informatisierung der Arbeit – Arbeitsforschung im Umbruch? Konzeptionelle Notwendigkeiten einer zukunftsfähigen Arbeitsforschung*. In: Baukowitz, A.; Berker, T.; Boes, A.; Pfeiffer, S.; Schmiede, R.; Will, M. (Hrsg.): *Informatisierung der Arbeit ? Gesellschaft im Umbruch*. Berlin: edition sigma.
- Böhle, F.; Bolte, A. (2002): *Die Entdeckung des Informellen. Der schwierige Umgang mit Kooperation im Arbeitsalltag*. Frankfurt/Main: Campus Verlag.
- Boltanski, L.; Chiapello, E. (2003): *Der neue Geist des Kapitalismus*. Konstanz: Universitätsverlag Konstanz.
- Boos, M. (1998): „Einer für alle“, „jeder für sich“ oder „mit den Augen des anderen“. *Führung und Zusammenarbeit in Gruppenentscheidungen*. In: Ardelt-Gattinger, E.; Lechner, H.; Schlögl, W. (Hrsg.), *Gruppendynamik. Anspruch und Wirklichkeit der Arbeit in Gruppen*, Göttingen: Verlag für Angewandte Psychologie, 113-127.<
- Boos, M., Jonas, K., Sassenberg, K. (2000, Hg.): *Computervermittelte Kommunikation in Organisationen*. Göttingen u.a.: Hogrefe.
- Borchert, M.; Urspruch, T. (2006): *Wer ist der IT-Freelancer? Ergebnisse der Umfrage unter IT-Freelancern vom 17. Oktober – 13. November 2005. Zusammenfassung des Untersuchungsberichts*. Lehrstuhl für Personal und Unternehmensführung, Department Management and Marketing, Universität Duisburg-Essen, Campus Duisburg, <http://www.vip-net.info>, 15.06.2006.
- Borum, F.; Westenholtz, A. (1995): *The incorporation of multiple institutional models. Organizational field multiplicity and the role of actors*. In: Scott, R.; Christensen, S. (Hrsg.): *The Institutional Construction of Organizations*, Thousand Oaks: Sage, 113-134.
- Bourdieu, P. (1987): *Die feinen Unterschiede. Kritik der gesellschaftlicher Urteilskraft*. Frankfurt/Main: Suhrkamp.
- Bourdieu, P. (1998): *Praktische Vernunft. Zur Theorie des Handelns*. Frankfurt/Main: Suhrkamp, 1985.
- Akerloff, G. A. (1970): *The Market for „Lemons“: Quality, Uncertainty and the Market Mechanism*. In: *Quarterly Journal of Economics* (84), 488-500.
- Bradach, J.; Eccles, R. (1991): *Price, authority and trust: from ideal types to plural forms*. In: Thompson, G.; Frances, J.; Levacic, R.; Mitchell, J. (Eds.): *Markets, hierarchies and networks. The coordination of social life*. London u.a.: Sage Publ., 277-292.
- Brand, A. (2002): *Funktion und Auftreten von Unternehmensnetzwerken aus der Sicht der Neuen Institutionenökonomie und soziologischer Ansätze. Welche Rolle spielt die Informationstechnik?* Stuttgart: IBIDEM-Verlag.
- Brand, A. (2004): *Die Rolle von Vertrauen bei Entstehung und Funktionsweise elektronischer Arbeitsnetze*. In: Hirschfelder, G.; Huber, B. (Hrsg.): *Die Virtualisierung der Arbeit. Zur Ethnographie neuer Arbeits- und Organisationsformen*. Frankfurt/Main: Campus Verlag. 165-198.
- Brand, A.; Holtgrewe, U. (2007): *KDE im Kontext: Open Source Software – Entwicklung und öffentliche Güter*. In: Moldaschl, M.; Stehr, N. (Hrsg.): *Knowledge Economy. Beiträge zur Ökonomie der Wissensgesellschaft*, Marburg: Metropolis. Im Erscheinen, s.a.: <http://soziologie.uni-duisburg.de/PERSONEN/holtgrewe/abuh-indsoz-604.pdf>, 15.04.2007.
- Brand, A.; Schmid, A. (2003): *Freiwillig zusammenarbeitende selbstständige Unternehmensberater. Fallstudie im Rahmen des DFG-Projekts Struktur und Funktionsweise elektronischer Arbeitsmärkte*, <http://www.soz.uni-frankfurt.de/arbeitslehre/pelm/docs/Nebenfallstudie%20freiw%20Arbeitsnetz%20Unternehmer.pdf>, 20.04.2007
- Brand, A.; Schmid, A. (2006): *Ist ein Open-Source-Projekt nur kooperativ? – Die Koordination der Zusammenarbeit im KDE-Projekt*. In: Lutterbeck, B.; Bärwolff, M.; Gehring, R. A. (Hrsg.): *Open Source Jahrbuch 2006. Zwischen Softwareentwicklung und Gesellschaftsmodell*. Berlin: Lehmanns Media, 123-137.
- Brand, K.-W. (1998): *Humanistischer Mittelklassen-Radikalismus. Die Erklärungskraft historisch-struktureller Deutungen am Beispiel der ‚neuen sozialen Bewegungen‘*. In: Hellmann, K.-U.; Koopmans, R. (Hrsg.): *Paradigmen der Bewegungsforschung. Entstehung und Entwicklung von Neuen sozialen Bewegungen und Rechtsextremismus*. Opladen: Westdeutscher Verlag, 33-50.

- Brödner, P. (2007): *Wissensteilung und Wissenstransformation*. In: Moldaschl, M.; Stehr, N. (Hrsg.): *Knowledge Economy. Beiträge zur Ökonomie der Wissensgesellschaft*, Marburg.
- Brown, J.; Duguid, P. (2001): *Knowledge and Organization: A Social-Practice Perspective*. In: *Organization Science*, 12, 198-213.
- Broy, M.; Schmidt, J. (1999): *Informatik: Grundlagenwissenschaft oder Ingenieurdisziplin?* In: *Informatik-Spektrum* 22, 206–209.
- Brucherseifer, E. (2004): *Die KDE-Entwicklergemeinde – wer ist das?* In: Gehring, R.; Lutterbeck, B. (Hrsg.): *Open Source Jahrbuch 2004. Zwischen Softwareentwicklung und Gesellschaftsmodell*. Berlin: Lehmanns Media, 65-82.
- Bruegger, Urs (1999): *Wie handeln Devisenhändler? Eine ethnographische Studie über Akteure in einem globalen Markt*. Bamberg: Difo-Druck OHG.
- Brunsson, N. (1989): *The Organization of Hipocrisy: Talk, decisions and Actions in Organizations*. Chichester: Wiley.
- Brynjolfsson, E.; Malone, T.; Gurbaxani, V. (1988): *Markets, Hierarchies and the Impact of Information Technology*. Working Paper, December, MIT Sloan School of Management.
- Bühl, W. L. (1974): *Einführung in die Wissenschaftssoziologie*; München: Beck.
- Burt, R. (1992): *Structural Holes. The social structure of competition*. Cambridge, MA: Harvard University Press.
- Busa, P.; Cunningham, R.; Ho, K.; McCullough, P. (1999): *“Free” pricing model. Can businesses really make money by giving goods away for free*. White Paper. University of Texas, Austin. <http://cci.mcombs.utexas.edu/research/white/free-price.htm>, 20.04.2007.
- Büschken, J. (1999): *Virtuelle Unternehmen – die Zukunft?* In: *DBW*, 59, 6, 778-791.
- Buschmeier, U. (1994): *Macht und Einfluß in Organisationen*. Göttingen: Cuvillier
- Butler, B.; Sproull, L.; Kiesler, S.; Kraut, R. (2002): *Community Effort in Online Groups: Who Does the Work and Why?* Working Paper, Forthcoming in: *Leadership at a Distance*, S. Weisband & L. Atwater (Eds.). February 5, 2002, v. 11.
- Büttemeyer, W. (1995): *Wissenschaftstheorie für Informatiker*. Heidelberg: Spektrum.
- c't (2004a): *Open-Source-Projekt einigt sich mit Fujitsu-Siemens über GPL-Rechte*. <http://www.heise.de/newsticker/meldung/45173>, 20.04.2007.
- c't (2004b): *Open-Source-Projekt einigt sich mit Router-Hersteller*, <http://www.heise.de/newsticker/meldung/44746>, 20.04.2007.
- Castells, M. (1996): *The rise of the network society*. Volume 1. Cambridge, MA: Blackwell Publishers.
- Claessens, D. (1983): *Die Gruppe unter innerem und äusserem Organisationsdruck*. In: *Kölner Zeitschrift für Soziologie und Sozialpsychologie*, 485-496.
- Claessens, D. (1999): *Gruppenverbände als Zusammenschluß kleiner Gruppen zu einer Großgruppe*. In: Schäfers, B. (Hrsg.): *Einführung in die Gruppensoziologie*. Wiesbaden: Quelle & Meyer, 3. Aufl. 157-176.
- Clemons, E.; Reddi, S.; Row, M. (1993): *The impact of information technology on the organization of economic activity: The „Move to the middle“ hypothesis*. In: *Journal of Management Information Systems*, Vol. 10, No. 2, Fall 1993, 9-35.
- Coase, R. (1937): *The nature of the firm*. In: Williamson, O.; Winter, S. (eds.): *The nature of the firm. Origins, Evolution, and Development*. Oxford u.a.: Oxford University Press, 1991.
- Cohendet, P.; Creplet, F.; Dupouët, O. (2001): *Organisational innovation, communities of practice and epistemic communities: the case of Linux*. In: Kirman, A.; Zimmermann, J. B. (eds.): *Economics with heterogeneous interacting agents*. Berlin: Springer.
- Coleman, J. S. (1991): *Grundlagen der Sozialtheorie, Band 1: Handlungen und Handlungssysteme*. München: Oldenbourg.
- Dackweiler, R.; Schäfer, R. (1999): *Lokal, national – international. Frauenbewegungspolitik im Rück- und Ausblick*. In: Klein, A.; Legrand, H.-J.; Leif, T. (Hrsg.): *Neue soziale Bewegungen. Impulse, Bilanzen und Perspektiven*. Opladen: Westdeutscher Verlag.
- Dahrendorf, R. (1961): *Homo Sociologicus*. Köln, Opladen: Westdeutscher Verlag.
- Della Porta, D.; Diani, M. (1999): *Social movements: an introduction*. Oxford: Blackwell Publishers.
- Demil, B.; Lecocq, X. (2003): *Neither Market nor hierarchy or network: The emerging bazaar governance*. Working Paper. <http://opensource.mit.edu/papers/demillecocq.pdf>, 20.04.2007.

- Dempsey, B.; Weiss, D.; Jones, P.; Greenberg, J. (1999): *A Quantitative Profile of a Community of Open Source Linux Developers*. SILS Technical Report TR-1999-05, School of Information and Library Science, University of North Carolina at Chapel Hill, Chapel Hill, North Carolina, <http://www.ibiblio.org/osrt/develop.html>, 20.04.2007.
- DiMaggio, P.; Powell, W. (1991a): *The iron cage revisited: Institutional isomorphism and collective rationality in organizational fields*. In: Powell, W.; DiMaggio, P. (Eds.): *The New Institutionalism in Organizational Analysis*. Chicago: Univ. of Chicago Press, 63-82, überarbeitete Version des gleichlautenden Artikels von 1983 in *American Sociological Review*, Vol. 48, April 1983, 147-160.
- DiMaggio, P.; Powell, W. (1991b): *Introduction*. In: Powell, W.; DiMaggio, P. (Eds.): *The New Institutionalism in Organizational Analysis*. Chicago: Univ. of Chicago Press, 1-38.
- Dolata, U. (1999): *Innovationsnetzwerke in der Biotechnologie?* In: WSI Mitteilungen, 2/1999, 132-141.
- Dolata, U. (2003): *Unternehmen Technik. Akteure, Interaktionsmuster und strukturelle Kontexte der Technikentwicklung: Ein Theorierahmen*. Berlin: edition sigma.
- Döring, N. (2003): *Sozialpsychologie des Internet. Die Bedeutung des Internet für Kommunikationsprozesse, Identitäten, soziale Beziehungen und Gruppen*. Göttingen: Hogrefe, 2. Aufl.
- Döring, N.; Schestag, A. (2000): *Eine empirische Analyse ausgewählter Chat-Channels*. In: Thiedecke, U. (Hrsg.): *Virtuelle Gruppen. Charakteristika und Problemdimensionen*. Wiesbaden: Springer, 313-355.
- Dose, C. (2005): *Flexible Bürokratie. Zur Logik aktueller Rationalisierungstendenzen bei Finanzdienstleistern*. Wiesbaden: Deutscher Universitäts-Verlag.
- Durkheim, E. (1977): *Über die Teilung der sozialen Arbeit*. Frankfurt/Main: Suhrkamp.
- Durkheim, E. (1984): *Die Regeln der soziologischen Methode*. Frankfurt/Main: Suhrkamp.
- Ebers, M.; Gotsch, W. (1999): *Institutionenökonomische Theorien der Organisation*. In: Kieser, A. (Hrsg.): *Organisationstheorien*. Stuttgart, Berlin, Köln: Kohlhammer, 3.Aufl., 199-252.
- Edeling, T. (1999): *Einführung: Der Neue Institutionalismus in Ökonomie und Soziologie*. In: Edeling, T.; Jann, W.; Wagner, D. (Hrsg.): *Institutionenökonomie und Neuer Institutionalismus. Überlegungen zur Organisationstheorie*. Opladen: Leske und Budrich Verlag.
- Edeling, T. (2002): *Organisationen als Institutionen*. In: Maurer, A.; Schmis M. (Hrsg.). *Neuer Institutionalismus. Zur soziologischen Erklärung von Organisation, Moral und Vertrauen*. Frankfurt/Main: Campus Verlag.
- Edelman, L.; Suchman, M. (1997): *The Legal Environments of organizations*. In: *Annual Review of Sociology*, Vol. 23, 479-515.
- Edge, D. (1990): *Competition in modern science*. In: Frängsmyr, T. (Ed.): *Solomon's house revisited. The organization and institutionalization of science*. Canton, MA: Science History Publications.
- Edwards, K. (2001): *"Epistemic communities, situated learning and F/OSS software development"*. Working paper, <http://opensource.mit.edu/papers/kasperedwards-ec.pdf>, 20.04.2007.
- Eggebrecht, A.; Flemming J.; Meyer G.; Müller A. v.; Oppolzer, A.; Paulinyi, A.; Schneider, H. (1980): *Geschichte der Arbeit. Vom Alten Ägypten bis zur Gegenwart*. Köln: Kiepenheuer & Witsch.
- Egloff, N. (1996): *Postindustrielle Dienstleistungsgesellschaft oder industrielle Arbeitsgesellschaft? Zum gesellschaftstheoretischen Kontext der These von der "Informationsgesellschaft"*. In: Schmiede, R. (Hrsg.): *Virtuelle Arbeitswelten. Arbeit, Produktion und Subjekt in der "Informationsgesellschaft"*. Berlin: Ed. Sigma Rainer Bohn Verlag.
- Elliott, M.; Scacchi, W. (2002). *Communicating and Mitigating Conflict in Open Source Software Development Projects*. Institute for Software Research, University of California, Irvine. <http://www.ics.uci.edu/~Emelliott/commossd.htm>, 20.04.2007.
- Elliott, M.; Scacchi, W. (2003). *Free Software: A Case Study of Software Development in a Virtual Organizational Culture*. Working Paper: Institute for Software Research, University of California Irvine, <http://www.ics.uci.edu/~wscacchi/Papers/New/Elliott-Scacchi-GNUE-Study-Report.pdf>, 20.04.2007.

- Erenkrantz, J. (2003): *Release Management within Open Source Projects*. In: Feller, J.; Fitzgerald, B.; Hissam, S.; Lakhani, K. (eds.): *Taking Stock of the Bazaar: Proceedings of the 3rd Workshop on Open Source Software Engineering*. 3rd Workshop on Open Source Software Engineering, ICSE'03, International Conference on Software Engineering, Portland, Oregon, May 3-11, 2003.
- Erlei, M.; Leschke, M.; Sauerland, D. (1999): *Neue Institutionenökonomik*. Stuttgart: Erlei Schäffer-Poeschel.
- Erlinghagen, M.; Knuth, M. (2002): *Auf der Suche nach dem "Turbo-Arbeitsmarkt", Zwischenbericht an die DFG zum Projekt "Restrukturierung des Arbeitsmarktes*. Graue Reihe des Instituts Arbeit und Technik 2002/03.
- Esposito, E. (1993): *Der Computer als Medium und Maschine*. In: *Zeitschrift für Soziologie*, 22: 338 – 354.
- Ettrich, M. (2004): *Koordination und Kommunikation in Open Source-Projekten*. In: Gehring, R.; Lutterbeck, B. (Hrsg.): *Open Source Jahrbuch 2004. Zwischen Softwareentwicklung und Gesellschaftsmodell*. Berlin: Lehmanns Media, 179-192.
- Fehr, E.; Gächter, S. (1998): *Reciprocity and economics: The economic implications of Homo Reciprocans*. In: *European Economic Review*, 42, 1998, 845-859.
- Feller, J.; Fitzgerald, B. (2000): *A Framework Analysis of the Open Source Software Development Paradigm*, Proc. 21st. Intern. Conf. Information Systems (ICIS), 58-69.
- Feller, J.; Fitzgerald, B. (2002): *Understanding Open Source Software Development*. London: Pearson Education.
- Flick, U. (1999): *Qualitative Forschung. Theorie, Methoden, Anwendung in Psychologie und Sozialwissenschaften*. Reinbek: Rowohlt, 1995, 4. Aufl.
- Fligstein, N. (1985): *The spread of the multidivisional form among large firms, 1919-1979*. In: *American Sociological Review*, Vol. 50, June 1985, 377-391.
- Fligstein, N. (1991): *The Structural Transformation of American Industry: An Institutional Account of the Causes of Diversification in the Largest Firms, 1919-1979*. In: Powell, W.; DiMaggio, P. (Eds.): *The New Institutionalism in Organizational Analysis*. Chicago: The University of Chicago Press: 311- 336.
- Franck, E.; Jungwirth, C. (2002a): *Das Open-Source-Phänomen jenseits des Gift-Society-Mythos*. In: *WiST*, H.3, März 2002.
- Franck, E.; Jungwirth, C. (2002b): *Reconciling investors and donators – The governance structure of open source*. Working Paper No. 8, Chair of Strategic Management and Business Policy, University of Zurich, June 2002; <http://opensource.mit.edu/papers/jungwirth.pdf>, 20.04.2007.
- Franke, N.; Shah, S. (2003): *How communities support innovative activities: an exploration of assistance and sharing among end-users*. In: *Research Policy* 32 (2003) 157–178.
- Freidson, E. (1994): *Professionalism Reborn. Theory, Prophecy and Policy*. Chicago: The University of Chicago Press.
- Freidson, E. (2001): *Professionalism. The third logic. On the practice of knowledge*. Chicago: The University of Chicago Press.
- French, J.; Raven, B. (1959): *The bases of social power*. In: Cartwright, D. (ed.), *Studies in social power* (150-167). Ann Arbor, MI: Institute for Social Research.
- Friedberg, E. (1995): *Ordnung und Macht. Dynamiken organisierter Handelns*. Frankfurt/Main; New York: Campus Verlag.
- Friedrich, J. (1992): *CASE-Tools und Software-Factories – Softwareentwicklung als Fabrikarbeit?* In: Trautwein-Kalms, G. (Hrsg.): *KontrastProgramm Mensch – Maschine. Arbeiten in der HighTech-Welt*. Köln.
- Fuchs-Heinritz, W.; Lautmann, R.; Rammstedt, O.; Wienold, H. (1995): *Lexikon zur Soziologie*. Opladen: Westdeutscher Verlag, 3.Aufl.
- Fuhse, J. (2002): *Kann ich dir vertrauen? Strukturbildung in dyadischen Sozialbeziehungen*. In: *ÖZP*, 31, 4
- Garud, R.; Kumaraswamy, A. (1995): *Coupling the technical and institutional faces of janus in network industries*. In: Scott, R.; Christensen, S. (Hrsg.): *The Institutional Construction of Organizations*, Thousand Oaks: Sage, 226-242.
- German, D. (2003): *The GNOME Project: a Case Study of Open Source, Global Software Development*. In: *Software process improvement and practice*, 8, 201-215.

- Ghosh, R. (1998): *Cooking-pot-markets: An economic model for the trade in free goods and services over the internet*. In: First Monday, Vol. 3, Nr. 3, [http://www.firstmonday.dk/issues/issue3\\_3/ghosh/](http://www.firstmonday.dk/issues/issue3_3/ghosh/), 20.02.2007.
- Ghosh, R.; David, P. (2003): *The nature and composition of the Linux kernel developer community: a dynamic analysis*. Draft version, <http://dxm.org/papers/licks1/licksresults.pdf>, 20.04.2007.
- Ghosh, R.; Glott, R.; Krieger, B.; Robles, G. (2002): *Free/Libre and Open Source Software: Survey and Study*. Deliverable D 18: Final Report, Part 4: Survey of Developers, Maastricht (International Institute of Infonomics), <http://www.infonomics.nl/FLOSS/report/>, 20.04.2007.
- Ghosh, R.; Prakash, V. (2000): *The Orbiten Free Software Survey*. In: First Monday, Vol. 5, Nr. 7, [http://www.firstmonday.org/issues/issue5\\_7/ghosh/](http://www.firstmonday.org/issues/issue5_7/ghosh/), 15.04.2007.
- Giddens, A. (1988): *Die Konstitution der Gesellschaft. Grundzüge einer Theorie der Strukturierung*. Frankfurt/Main: Campus Verlag.
- Glock, F. (1997): *Zur Soziologie des Konstruierens. Arbeitspapier*, WZ-Berlin. FS II-97-114.
- Göbel, E. (2002): *Neue Institutionenökonomik. Konzeption und betriebswirtschaftliche Anwendungen*. Stuttgart: Lucius und Lucius.
- Goll, M. (2002): *Arbeiten Im Netz. Kommunikationsstrukturen, Arbeitsabläufe, Wissensmanagement*. Wiesbaden: VS Verlag für Sozialwissenschaften.
- González-Barahona, J.; Robles-Martínez, G. (2003): *Unmounting the "code gods" assumption*. <http://libresoft.urjc.es/Papers/InfoPaper?id=unmountingthe%22codegod%22assumption>, 20.04.2007.
- Götzenburger, G.; Löger, B. (2000): *Online Communities. Struktur sozialer Beziehungen und Spielermotivationen am Beispiel von Multi User Dungeons*. In: Thiedecke, U. (Hrsg.): *Virtuelle Gruppen. Charakteristika und Problemdimensionen*. Wiesbaden: Springer.
- Granados, G.; Gurgsdies, E. (1999): *Lern- und Arbeitsbuch Ökonomie*. Bonn: Dietz.
- Grasmuck, V. (2002): *Freie Software zwischen Privat- und Gemeineigentum*. Bonn: Bundeszentrale für politische Bildung.
- Gruber, M.; Henkel, J. (2004): *New ventures based on open innovation – an empirical analysis of start-up firms in embedded Linux*. Working paper, University of Munich, July 2004, <http://opensource.mit.edu/papers/gruberhenkel.pdf>; 20.04.2007.
- Grunwald, W.; Lilje, H. (1981, Hrsg.): *Kooperation und Konkurrenz in Organisationen*. Bern: Haupt.
- Gukenbiehl, H. (1999): *Bezugsgruppen*. In: Schäfers, B. (1999; Hrsg.): *Einführung in die Gruppensoziologie*. Wiesbaden: Quelle & Meyer, 3. Aufl., 113-134.
- Gukenbiehl, H. L. (1993): *Institution und Organisation*. In: Korte, H.; Schäfers, B. (Hrsg.): *Einführung in die Hauptbegriffe der Soziologie*, 2. Aufl., Opladen: Leske und Budrich, 95-110.
- Gukenbiehl, H.; Kopp, J. (2003): *Macht*. In: Schäfers, B. (Hrsg.): *Grundbegriffe der Soziologie*. Opladen: Leske und Budrich Verlag, 8. Aufl., 210-211.
- Habermas, J. (1971): *Vorbereitende Bemerkungen zu einer Theorie der kommunikativen Kompetenz*. In: Habermas, J.; Luhmann, N. (Hrsg.): *Theorie der Gesellschaft oder Sozialtechnologie. Was leistet die Systemforschung?* Frankfurt/Main: Suhrkamp, 101-141.
- Habermas, J. (1997): *Theorie des kommunikativen Handelns. Band 1. Handlungsrationalität und gesellschaftliche Rationalisierung*. Frankfurt/Main: Suhrkamp, 2. Aufl., erstmals 1981
- Hagstrom, W. (1982): *Gift giving as an organizing principle in science*. In: Barnes B.; Edge, D. (Eds): *Science in Context: Readings in the Sociology of Science*. Cambridge, Mass.: MIT Press.
- Håkanson, L. (2002): *"Creating knowledge – the power and logic of articulation" (what the fuss is all about)*. Paper presented for the LINK conference, November 1-2.
- Hasse, R.; Krücken, G. (1999): *Neo-Institutionalismus*. Bielefeld: transcript Verl.
- Häußermann, H.; Siebel, W. (1995): *Dienstleistungsgesellschaften*. Frankfurt/Main: Suhrkamp.
- Haythornthwaite, C.; Wellmann, B.; Garton, L. (2000): *Arbeit und Gemeinschaft bei computervermittelter Kommunikation*. In: Thiedecke, U. (Hrsg.): *Virtuelle Gruppen. Charakteristika und Problemdimensionen*. Wiesbaden: Springer, 356-392.
- Heidegger, M. (1982): *Die Technik und die Kehre*. 5. Aufl., 1962, Pfullingen: Neske.
- Heidenreich, M. (2000): *Regionale Netzwerke in der globalen Wissensgesellschaft*. In: Weyer, J. (Hrsg.): *Soziale Netzwerke*. München u.a.: Oldenbourg, 87-110.

- Heider, F.; Mevissen, M. (1991): *Selbstverwaltete Betriebe in Hessen: Eine sozio-ökonomische Analyse unter besonderer Berücksichtigung der Situation der Frauen*. Giessen: Focus.
- Heidling, E. (2000): *Strategische Netzwerke. Koordination und Kooperation in asymmetrisch strukturierten Unternehmensnetzwerken*. In: Weyer, J. (Hrsg.): *Soziale Netzwerke*. München u.a.: Oldenbourg, 63-85.
- Heintz, B. (1993): *Die Herrschaft der Regel. Zur Grundlagengeschichte des Computers*. Frankfurt/Main: Campus Verlag.
- Heintz, B. (2000a): *Die Innenwelt der Mathematik. Zur Kultur und Praxis einer beweisenden Disziplin. Ästhetik und Naturwissenschaften, Bildende Wissenschaften – Zivilisierung der Kulturen*. Wien: Springer.
- Heintz, B. (2000b): *Gemeinschaft ohne Nähe? Virtuelle Gruppen und reale Netze*. In: Thiedecke, U. (Hrsg.): *Virtuelle Gruppen. Charakteristika und Problemdimensionen*. Wiesbaden: Springer, 188-218.
- Heinze, R.; Offe C. (1990; Hrsg.): *Formen der Eigenarbeit: Theorie, Empirie, Vorschläge*. Opladen: Westdeutscher Verlag.
- Hellmann, K.-U. (1998): *Paradigmen der Bewegungsforschung. Forschungs- und Erklärungsansätze – ein Überblick*. In: Hellmann, K.-U.; Koopmans, R. (Hrsg.): *Paradigmen der Bewegungsforschung. Entstehung und Entwicklung von Neuen sozialen Bewegungen und Rechtsextremismus*. Opladen: Westdeutscher Verlag, 9-32.
- Henninger, A.; Sieber, A. (2001): *Softwareentwicklung in kleinen Unternehmen in Ost- und Westdeutschland*. In: Matuschek, I.; Henninger, A.; Kleemann, F. (Hrsg.): *Neue Medien im Arbeitsalltag: Empirische Befunde – Gestaltungskonzepte – Theoretische Perspektiven*. Opladen: Westdeutscher Verlag, 37-54.
- Herlyn, I. (1999): *Gruppen in schulischen Lehr- und Lernprozessen*. In: Schäfers, Bernhard (Hrsg.): *Einführung in die Gruppensoziologie*. Wiesbaden: Quelle & Meyer, 3. Aufl. 157-176.
- Hertel, G.; Niedner, S.; Herrmann, S. (2003): *Motivation of software developers in open source projects: An internet-based survey of contributors to the Linux kernel*. In: *Research Policy* 32 (7), 1159-1177.
- Heun, W. (1983): *Das Mehrheitsprinzip in der Demokratie. Grundlagen – Strukturen – Begrenzungen*. Berlin: Duncker und Humblot.
- Hillmann, K. (1994): *Wörterbuch der Soziologie*. Stuttgart: Kröner, 4.Aufl.
- Hinner, K. (2003): *Wissenschaft im Zeitalter des Internet*. Berlin: Logos Verlag.
- Hinrichs, K. (1989): *Geld und Zeit in privaten Haushalten: Gelegenheitsstruktur und Bedarf für Eigenarbeit als Determinanten sozialer Ungleichheit*. Bielefeld: AJZ-Druck und Verlag.
- Hippel, E. v. (2001): *Open Source Shows the Way: Innovation by and for Users – No Manufacturer Required!* In: *Sloan Management Review*, <http://opensource.mit.edu/papers/evhippel-osuserinnovation.pdf>, 20.04.2007.
- Hippel, E. v.; Krogh, G. v. (2003): *Open Source Software and the Private-Collective Innovation Model: Issues for Organization Science*. Working Paper, <http://opensource.mit.edu/papers/hippelkrogh.pdf>, 20.04.2007.
- Hirsch, P. (1997): *Review Essay: Sociology without social structure: Neoinstitutional theory meets brave new world*. In: *American Journal of Sociology*, 102, 1702-1723.
- Hirsch-Kreinsen, H. (1995): *Dezentralisierung: Unternehmen zwischen Stabilität und Desintegration*. In: *Zeitschrift für Soziologie*, Jg. 24, H.6, Dez. 1995, 422-435.
- Hirsch-Kreinsen, H. (2002): *Unternehmensnetzwerke – revisited*. In: *Zeitschrift für Soziologie*, Jg. 31, H.2, April 2002, 106-124.
- Hoegner, W. (2006): *Das Projekt LiMux – Freie Software für die Münchner Verwaltungsclients*. In: Gehring R. A.; Lutterbeck B. (Hrsg.): *Open Source Jahrbuch 2006. Zwischen Softwareentwicklung und Gesellschaftsmodell*. Berlin: Lehmanns Media.
- Hoffmann, E.; Walwei, U. (1998): *Normalarbeitsverhältnis: Ein Auslaufmodell? – Überlegungen zu einem Erklärungsmodell für den Wandel der Beschäftigungsformen*. In: *MittAB*, 31. Jg., Heft 3, 409-425.
- Hoffmann; U. (2000): *Neues vom Baron Münchhausen. Die institutionelle Selbstorganisation bei der Bildung virtueller Gruppen im Usenet*. In: Thiedecke, U. (Hrsg.): *Virtuelle Gruppen. Charakteristika und Problemdimensionen*. Wiesbaden: Springer, 168-187.
- Holler, M.J.; Illing, G. (1996): *Einführung in die Spieltheorie*, 3. Aufl. Berlin: Springer.

- Hollingsworth, J.; Boyer, R. (1997): *Coordination of Economic Actors and Social Systems of Production*. In: Hollingsworth, J.; Boyer, R. (Ed.): *Contemporary Capitalism. The Embeddedness of Institutions*. Cambridge: Cambridge Univ. Press, 1-47.
- Holtgrewe, U. (2004a): *Articulating the Speed(s) of the Internet: The Case of Open Source/Free Software*. In: *Time & Society* 13: 129-46.
- Holtgrewe, U. (2004b): *Heterogene Ingenieure – Open Source und Freie Software zwischen technischer und sozialer Innovation*. In: Gehring, R.; Lutterbeck, B. (Hrsg.): *Open Source Jahrbuch 2004. Zwischen Softwareentwicklung und Gesellschaftsmodell*. Berlin: Lehmanns Media, 339-351.
- Holtgrewe, U. (2005): *Werkzeuge, Kooperationen und Institutionen erfinden – Open Source-Softwareentwicklung als technische/soziale Innovation*. In: Aderhold, J.; John, R.: *Soziale Innovation (Arbeitstitel)*. Konstanz (UVK)-
- Holtgrewe, U.; Brand, A. (2007): *Die Projektpolis bei der Arbeit. Open Source Software Entwicklung und der „neue Geist des Kapitalismus“*. Erscheint in *Öster. Zeitschr. f. Soziologie*, 3/2007.
- Holtgrewe, U.; Werle, R. (2001): *De-Commodifying Software? Open Source Software Between Business Strategy and Social Movement*. In: *Science Studies*, 2/2001.
- Homann, K.; Suchanek, A. (2005): *Ökonomik: Eine Einführung*, 2. Aufl. Tübingen: Mohr Siebeck.
- Hornecker, E. (1998): *Programmieren als Handwerkszeug im ersten Semester*. In: Claus, V. (Hrsg.) *Informatik und Ausbildung, GI-Fachtagung '98*, 30.3. - 1.4.1998, Stuttgart. Springer, *Informatik Aktuell* 1998, 43-51.
- Hradil, S. (1980): *Die Erforschung der Macht: Eine Übersicht über die empirische Ermittlung von Machtverteilungen durch die Sozialwissenschaften*. Stuttgart: Kohlhammer.
- Huber, B.; Hirschfelder, G. (2004): *Neue Medien und Arbeitswelt – zur Einführung*. In: Hirschfelder, G.; Huber, B. (Hrsg.): *Die Virtualisierung der Arbeit. Zur Ethnographie neuer Arbeits- und Organisationsformen*. Frankfurt/Main: Campus Verlag. 11-26.
- Huber, L. (1991): *Sozialisation in der Hochschule*. In: Hurrelmann K.; Ulich, D. (Hrsg.): *Neues Handbuch der Sozialisationsforschung*. Weinheim.
- Hubig, C. (2004): *Technik als Mittel und als Medium*. In: Karafyllis, N.; Haar, T. (Hrsg.): *Technikphilosophie im Aufbruch. Festschrift für Günter Ropohl*. Berlin: Ed. Sigma, 95-109.
- Iannacci, F. (2003): *The Linux managing model*. In: *First Monday*, Vol. 8, Nr. 12, [http://firstmonday.org/issues/issue8\\_12/iannacci](http://firstmonday.org/issues/issue8_12/iannacci), 20.04.2007.
- Jaeger, T.; Metzger, A. (2002): *Open Source Software. Rechtliche Rahmenbedingungen der Freien Software*. München: Beck.
- Jansen, D. (1999): *Einführung in die Netzwerkanalyse. Grundlagen, Methoden, Anwendungen*. Opladen: Leske und Budrich Verlag.
- Jansen, S. (2000): *Konkurrenz der Konkurrenz! Co-opetition: Die Form der Konkurrenz – Typen, Funktionen und Voraussetzungen von paradoxen Koordinationsformen*. In: Jansen, S.; Schleissing, S. (Hrsg.): *Konkurrenz und Kooperation. Interdisziplinäre Zugänge zur Theorie der Co-opetition*. Marburg: Metropolis.
- Jones, C.; Hesterly, W.; Borgatti, S. (1997): *A general theory of network governance: Exchange conditions and social mechanisms*. In: *Academy of Management Review*, Vol.22, No.4, 911-945.
- Jørgensen, N. (2001): *Putting it all in the trunk: incremental software development in the FreeBSD open source project*. In: *Information Systems Journal* (2001), 11, No. 4, 321–336
- Jung, H. (2005): *Personalwirtschaft*. München: Oldenbourg, 6. Aufl.
- Jütting, D.; Bentem, N. v.; Oshege, V. (2003): *Vereine als sozialer Reichtum. Empirische Studien zu lokalen freiwilligen Vereinigungen*. Münster: Waxmann.
- Kautz, K.; Ramzan, F. (2001): *Software Quality Management and Software Process Improvement in Denmark*. In: *Proceedings of the 34th Hawaii International Conference on System Sciences – 2001*,
- Kawai, A.; Schmid, A. (2004): *Beschäftigungsflexibilität durch interne Jobbörsen. Wettbewerb und Hierarchie auf unternehmensinternen elektronischen Arbeitsmärkten*. In: Struck, O.; Köhler, C. (Hrsg.), *Beschäftigungsstabilität im Wandel*, München und Merkung, 247-265.
- Kelty, C. M. (2001): *Free Software/Free Science*. In: *First Monday*, Vol. 6, Nr. 12, [http://www.firstmonday.org/issues/issue6\\_12/kelty/](http://www.firstmonday.org/issues/issue6_12/kelty/), 20.04.2007.

- Kieser, A.; Kubicek, H. (1992): *Organisation*. Berlin: de Gruyter, 3. Aufl.
- Klatt, R. (1999): *Wie das Leitbild des virtuellen Unternehmens wirkt. Beispiele aus der Unternehmenspraxis*. In: Arbeit, H.3, Jg.8, 241-254.
- Klein, H. (2001): *Vereine*. In: Schäfers, B.; Zapf, W. (Hrsg.): Handwörterbuch zur Gesellschaft. Bonn: Bundeszentrale für politische Bildung, 2. neubearb. Aufl.
- Kliment, T. (1998): *Durch Dramatisierung zum Protest? Theoretische Grundlegung und empirischer Ertrag des Framing-Konzepts*. In: Hellmann, K.-U.; Koopmans, R. (Hrsg.): Paradigmen der Bewegungsforschung. Entstehung und Entwicklung von Neuen sozialen Bewegungen und Rechtsextremismus. Opladen: Westdeutscher Verlag, 69-89.
- Knorr-Cetina, K. (2002): *Wissenskulturen. Ein Vergleich naturwissenschaftlicher Wissensformen*. Frankfurt/Main: Suhrkamp, zuerst engl. 1999.
- Koch, S.; Schneider, G. (2002): *Effort, co-operation and co-ordination in an open source software project: GNOME*. In: Information Systems Journal (2002) 12, 27-42.
- Kollmann, T. (2001): *Virtuelle Marktplätze: Grundlagen – Management – Fallstudie*. München: Vahlen.
- Kollock, P. (1999): *The economies of online cooperation: Gifts and public goods in cyberspace*. In: Smith, M.; Kollock, P. (Eds.): Communities in Cyberspace. London: Routledge
- Konrad, W.; Gerd P. (1999): *Innovation in der Softwareindustrie. Organisation und Entwicklungsarbeit*. Frankfurt/Main: Campus Verlag.
- Kortzfleisch, H. (1999): *Virtuelle Unternehmen. Sammelrezension*. In: Die Betriebswirtschaft, 59, 1999, 5, 664-685.
- Kotthoff, H.; Reindl, J. (1990): *Die soziale Welt kleiner Betriebe. Wirtschaften, Arbeiten und Leben im mittelständischen Industriebetrieb*. Göttingen: Krebs.
- Kowol, U.; Krohn, W. (1997): *Modernisierungsdynamik und Innovationslethargie*. In: Blätzel-Mink, B.; Renn, O. (Hrsg.): Zwischen Akteur und System. Die Organisierung von Innovation. Opladen: Westdeutscher Verlag, 39-65.
- Kowol, Uli (1998): *Innovationsnetzwerke: Technikentwicklung zwischen Nutzungsvisionen und Verwendungspraxis*. Wiesbaden: Deutscher Universitätsverlag.
- Krebs, M.; Rock, R. (1994): *Unternehmensnetzwerke – eine intermediäre oder eigenständige Form?* In: Sydow, J.; Windeler, A. (Hrsg.): Management interorganisationaler Beziehungen. Opladen: Westdeutscher Verlag, 322-345.
- Krishnamurthy, S. (2002): *Cave or Community? An Empirical Examination of 100 Mature Open Source Projects*. Working Paper, University of Washington, Bothell, <http://opensource.mit.edu/papers/krishnamurthy.pdf>, 20.04.2007.
- Krol, G.; Schmid, A. (2002): *Volkswirtschaftslehre. Eine problemorientierte Einführung*. Tübingen: Mohr Siebeck; 21. Aufl.
- Krömmelbein, S. (2004): *Kommunikativer Stress in der Arbeitswelt. Zusammenhänge von Arbeit, Interaktion und Identität*. Berlin: Ed. Sigma.
- Krömmelbein, S.; Schmid, A. (2000): *Zur Struktur und Funktionsweise von Arbeitsmarktnetzwerken*. In: Schmid, A.; Krömmelbein, S. (Hrsg.): Globalisierung, Vernetzung und Erwerbsarbeit. Theoretische Zugänge und empirische Entwicklungen. Wiesbaden: Dt. Univ. Verlag, 215-243.
- Krömmelbein, S.; Schmid, A. (2004): *Informationstechnologischer Wandel und Zukunft der Arbeit*. In: Karafyllis, N.; Haar, T. (Hrsg.): Technikphilosophie im Aufbruch. Festschrift für Günter Ropohl. Berlin: Ed. Sigma, 211-227.
- Krücken, G. (2004): *Hochschulen im Wettbewerb – eine organisationstheoretische Perspektive*. In: Böttcher, W.; Terhardt, E. (Hrsg.): Organisationstheorie: Ihr Potential für die Analyse und Entwicklung von pädagogischen Feldern. Wiesbaden: VS Verlag für Sozialwissenschaften.
- Kühl, S. (1999): *Krise, Renaissance oder Umbau von Hierarchien in Unternehmen? Anmerkungen zur aktuellen Managementdiskussion*. In: Berliner Debatte Initial, 10 (1999), 3.
- Kumbruck, C. (2000): *Digitale Signaturen und Vertrauen*. In: Arbeit, H.4, Jg.9, 105-118.
- Kumbruck, C. (2001): *Was ist Kooperation? Kooperation im Lichte der Tätigkeitstheorie*. In: Arbeit, H.2, Jg. 10, 149-166.
- Kuwabara, K. (2000): *Linux: a bazaar at the edge of chaos*. In: First Monday, Vol. 5, Nr. 3. [http://firstmonday.org/issues/issue5\\_3/kuwabara/](http://firstmonday.org/issues/issue5_3/kuwabara/), 20.04.2007.

- Lakhani, K. R.; Wolf, B. (2003): *Why Hackers Do What They Do: Understanding Motivation and Effort in Free/Open Source Software Projects*. <http://opensource.mit.edu/papers/lakhaniwolf.pdf>, 20.04.2007.
- Lane, C.; Bachmann, R. (1996): *The social constitution of trust: Supplier relations in Britain and Germany*. In: *Organization Studies*, 17, 3.
- Lant, T.; Baum, J. (1995): *Cognitive Sources of Socially Constructed Competitive Groups: Examples From the Manhattan Hotel Industry*. In: Scott, R.; Christensen, S. (Hrsg.): *The Institutional Construction of Organizations*, Thousand Oaks: Sage, 15-38.
- Latour, B.; Woolgar, S. (1979): *Laboratory Life: The construction of scientific facts*. Princeton, New Jersey, 2. Aufl. 1986/1979, 235-261.
- Laudel, G. (2001). *Collaboration, creativity and rewards: why and how scientists collaborate*. *International Journal of Technology Management*, 22, 762 -781.
- Laudel, G.; Gläser, J. (1998): *What are Institutional Boundaries and how can They be Overcome? Germany's Collaborative Research Centres as Boundary-Spanning Networks*. Veröffentlichungsreihe der Arbeitsgruppe Wissenschaftstransformation des Wissenschaftszentrums Berlin für Sozialforschung, Arbeitspapier Nr. P 98-401, Berlin, September 1998.
- Laux, H.; Liermann, F. (2003): *Grundlagen der Organisation. Die Steuerung von Entscheidungen als Grundproblem der Betriebswirtschaftslehre*. Heidelberg: Springer, 5. Aufl.
- Lazega, E. (2001): *The collegial phenomenon: The social mechanisms of cooperation among peers in a corporate law partnership*. Oxford, UK: Oxford University Press.
- Leicht, R.; Philipp, R. (1999): *Der Trend zum Ein-Personen-Unternehmen. Mehr Selbständige: Abschied vom "Normal-Unternehmer"?* In: Strukturbericht des Instituts für Mittelstandsforschung, Heft 5, Mannheim.
- Leiteritz, R. (2002): *Der kommerzielle Einsatz von Open Source Software und kommerzielle Open Source-Geschäftsmodelle. Zur Bedeutung von Open Source Software in Unternehmen und als Grundlage für Geschäftsmodelle*. Diplomarbeit an der Technischen Universität Berlin, Fachbereich Informatik.
- Leiteritz, R. (2003): *Microsoft – Geschäftsmodell und Erfolgsfaktoren*. [http://ig.cs.tu-berlin.de/oldstatic/ap/rl/2003-01/microsoft\\_geschaeftsmodell\\_erfolgsfaktoren.pdf](http://ig.cs.tu-berlin.de/oldstatic/ap/rl/2003-01/microsoft_geschaeftsmodell_erfolgsfaktoren.pdf), 20.04.2007.
- Lerner, J.; Tirole, J. (2000): *The Simple Economics of Open Source*. NBER Working Paper 7600. <http://www.people.hbs.edu/jlerner/simple.pdf>, 20.04.2007.
- Lerner, J.; Tirole, J. (2002): *The Scope of Open Source Licensing. November 19, 2002*, In: <http://opensource.mit.edu/papers/lernertirole2.pdf>, 20.04.2007.
- Leuze, K. (2007): *Zwischen Mythos und Mikropolitik: Soziale Kompetenz in der Wissensarbeit*. In: Moldaschl, Manfred/Steher, Nico (Hg.): *Knowledge Economy. Beiträge zur Ökonomie der Wissensgesellschaft*, Marburg: Metropolis. Im Erscheinen
- Levine, J.; Moreland, R. (1995): *Group processes*. In: Tesser, Abraham (Hrsg.): *Advanced Social Psychology*. New York: McGraw-Hill.
- Levine, S. (2001): *Kindness in Cyberspace? The sharing of valuable goods on-line*. In: Gattiker, U. (Ed.): *Conference Proceedings EICAR International Conference*, 86-112.
- Liebold, R.; Trinczek, R. (2002): *Experteninterview*. In: Kühl, S.; Strodtholz, P. (Hrsg.): *Methoden der Organisationsforschung. Ein Handbuch*. Reinbek: Rowohlt, 33-70.
- Lilienthal, M. (2001): *Durkheim: Über soziale Arbeitsteilung*. In: Gamm, G.; Hetzel, A.; Lilienthal M. (Hrsg.): *Hauptwerke der Sozialphilosophie*. Stuttgart: Reclam, 57-71.
- Ljungberg, J. (2000): *F/OSS movements as a model for organizing*. In: *European Journal of Information Systems*, 94, pp. 208-216
- Lo, V. (2001): *Zwischen regionalen und globalen Beziehungen: Wissensbasierte Netzwerke im Finanzsektor*. In: Esser, J.; Schamp, E. (Hrsg.): *Metropolitane Region in der Vernetzung. Der Fall Frankfurt/Rhein-Main*. Frankfurt/Main: Campus Verlag.
- Luhmann, N. (1987): *Soziale Systeme. Grundriß einer allgemeinen Theorie*. Frankfurt/Main: Suhrkamp.
- Luhmann, N. (1990): *Die Wissenschaft der Gesellschaft*. Frankfurt/Main: Suhrkamp.

- Luthiger, B. (2004): *Alles aus Spaß? Zur Motivation von Open-Source-Entwicklern*. In: Gehring R.; Lutterbeck, B. (Hrsg.): *Open Source Jahrbuch 2004. Zwischen Softwareentwicklung und Gesellschaftsmodell*. Berlin: Lehmanns Media, 93-106.
- Machwirth, E. (1999): *Die Gleichaltrigengruppe (peer-group) der Kinder und Jugendlichen*. In: Schäfers, B. (Hrsg.): *Einführung in die Gruppensoziologie*. Wiesbaden: Quelle & Meyer, 3. Aufl., 248-268.
- Mahnkopf, B. (1994): *Markt, Hierarchie und soziale Beziehungen. Zur Bedeutung reziproker Beziehungsnetzwerke in modernen Marktgesellschaften*. In: Beckenbach, N.; Treeck, W. van (Hrsg.): *Umbrüche gesellschaftlicher Arbeit*. Göttingen: Schwartz, Soziale Welt Sonderband 9.
- Malone, T.; Yates, J.; Benjamin, R. (1987): *Electronic markets and electronic hierarchies*. In: *Communications of the ACM*, Vol.30, No.6, June 1987, 484-497.
- Marcuse, H. (1994): *Der eindimensionale Mensch. Studien zur Ideologie der fortgeschrittenen Industriegesellschaft*. Frankfurt/Main: Suhrkamp.
- Marin, B.; Mayntz, R. (1991, Hrsg.): *Policy Networks. Empirical Evidence and Theoretical Considerations*, Frankfurt/Main: Boulder, Co.
- Mark, G. (2002): *Conventions and Commitments in Distributed Groups*. In: *Computer Supported Cooperative Work – The Journal of Collaborative Computing*, vol. 11, no. 3-4, pp. 349-387.
- Marrs, K.; Boes, A. (2003): *Alles Spaß und Hollywood? Arbeits- und Lebensbedingungen bei Film und Fernsehen*. In: Pohlmann, M.; Sauer, D.; Trautwein-Kalms, G.; Wagner, A. (Hrsg.): *Dienstleistungsarbeit: Auf dem Boden der Tatsachen. Befunde aus Handel, Industrie, Medien und IT-Branche*. Edition Sigma, Berlin.
- Mauss, M. (1984): *Die Gabe: Form und Funktion des Austauschs in archaischen Gesellschaften*. Frankfurt/Main: Suhrkamp.
- Mayntz, R. (1987): *Politische Steuerung und gesellschaftliche Steuerungsprobleme – Anmerkungen zu einem theoretischen Paradigma*. In: *Jahrbuch zur Staats- und Verwaltungswissenschaft B1/1987*, 89-110.
- Mayntz, R. (1992): *Modernisierung und die Logik von interorganisatorischen Netzwerken*. In: *Journal für Sozialforschung*, 3, 2, 19-31.
- McAdam, D.; McCarthy, J.; Mayer, Z. (1988): *Social Movements*. In: Smelser, N. (Ed.): *Handbook of sociology*. Newbury Park, CA: Sage Publ., 695-737.
- Meeker, B. F. (1971): *Decisions and Exchange*. In: *American Sociological Review*. Vol. 36, No. 3, (Jun., 1971), 485-495.
- Menard, C. (2004): *The economics of hybrid organizations*. In: *Journal of institutional and theoretical economics*. 160 (2004), 345-376.
- Merton, R. (1985): *Entwicklung und Wandel von Forschungsinteressen. Aufsätze zur Wissenschaftssoziologie*. Frankfurt/Main: Suhrkamp, erstmals engl. 1973.
- Messner, D. (1994): *Fallstricke und Grenzen der Netzwerksteuerung*. In: *Prokla*, H.97, 24. Jg., Nr.4, 563-596.
- Meyer, J.; Rowan, B. (1991): *Institutionalized organisations: Formal structure as myth and ceremony*. In: Powell, W.; DiMaggio, P. (Eds.): *The new institutionalism in organizational analysis*. Chicago: Univ. of Chicago Press, erstmals 1977 in *American Journal of Sociology*, 83, 340-363.
- Meyer, P. (2003): *Episodes of collective invention*. Working Paper 368 of Bureau of Labor Statistics, August 2003, <http://opensource.mit.edu/papers/meyer.pdf>, 20.04.2007.
- Miegel, M.; Wahl, S. (2002): *Arbeitsmarkt und Arbeitslosigkeit. Probleme und Perspektiven*. München: Olzog.
- Milgrom, P.; Roberts, J. (1992): *Economics, Organization, and Management*. Englewood Cliffs: Prentice Hall.
- Mizuchri, M.; Fein, L. (1999): *The social construction of organizational knowledge. A study of the uses of coercive, mimetic, and normative isomorphism*. In: *Administrative Science Quarterly*, Vol. 44, No. 4, 653-683.
- Mockus, A.; Fielding, R.; Herbsleb, J. (2002): *Two Case Studies of Open Source Software Development: Apache and Mozilla*. In: *ACM Trans. Software Engineering and Methodology*, 11(3), 309-346, 2002, <http://citeseer.ist.psu.edu/mockus02two.html>, 20.04.2007.

- Moldaschl, M.; Sauer, D. (2000): *Internalisierung des Marktes – Zur neuen Dialektik von Kooperation und Herrschaft*. In: Minssen, H. (Hrsg.): *Begrenzte Entgrenzungen: Wandlungen von Organisation und Arbeit*. Berlin: Ed. Sigma.
- Möllering, G. (2001): *The Nature of Trust: From Georg Simmel to a Theory of Expectation, Interpretation and Suspension*. In: *Sociology*, Vol. 35, No. 2, pp. 403–420.
- Moon, J.; Sproul, L. (2000): *Essence of distributed work: The case of the linux kernel*. In: *First Monday*, Vol. 5, Nr. 11, [http://firstmonday.org/issues/issue5\\_11/moon/](http://firstmonday.org/issues/issue5_11/moon/), 20.04.2007.
- Müller, T. (1997): *Virtuelle Organisation. Konzept, Theoriebasis, Möglichkeiten und Grenzen*. In: *Management Forschung und Praxis*, Heft 21, Oktober 1997, Konstanz.
- Müller-Jentsch, W. (2003): *Organisationssoziologie. Eine Einführung*. Frankfurt/Main: Campus Verlag.
- Multrus, F. (2004): *Fachkulturen. Begriffsbestimmung, Herleitung und Analysen. Eine empirische Untersuchung über Studierende deutscher Hochschulen*. Dissertation an der Universität Konstanz.
- Murtha, T. P.; Lenway, S. A.; Hart, J. A. (2001): *Managing New Industry Formation: Global Knowledge Formation and Entrepreneurship in High Technology*. Stanford: Stanford University Press.
- Mutz, G. (2002): *Pluralisierung und Entgrenzung in der Erwerbsarbeit, im Bürgerengagement und in der Eigenarbeit*. In: *Arbeit*, H.14, Jg. 11 (2002), 21-32.
- Narduzzo, A.; Rossi, A. (2003): *Modularity in Action: GNU/Linux and Free/Open Source Software Development Model Unleashed*. Working Paper, University of Bologna, Quaderno DISA n. 78, May 12, 2003.
- Nave-Herz, R. (1993): *Die Geschichte der Frauenbewegung in Deutschland*. Hannover: Niedersächsische Landeszentrale für politische Bildung, 4. Aufl.
- Neidhardt, F. (1999): *Das „Innere System“ sozialer Gruppen und ihr Außenbezug*. In: Schäfers, B. (Hrsg.): *Einführung in die Gruppensoziologie*. Wiesbaden: Quelle & Meyer, 3. Aufl., 135-158.
- Nett, B.; Wulf, V. (2005): *Kleine und mittelständische Unternehmen unter empirischer Perspektive. Wissensprozesse in der Softwarebranche*. In: Gendolla, P.; Schäfer, J. (Hrsg.): *Wissensprozesse in der Netzwerkgesellschaft*, transcript Verlag, Bielefeld, 147-168.
- Neuberger, O. (1998): *Strategische Kooperation (Mikropolitik)*. In: Spieß, E. (Hrsg.): *Formen der Kooperation – Bedingungen und Perspektiven*. Göttingen: Hogrefe, 37-52.
- Niechoj, T. (2003): *Kollektive Akteure zwischen Wettbewerb und Steuerung. Effizienz und Effektivität von Verhandlungssystemen aus ökonomischer und politikwissenschaftlicher Sicht*. Marburg: Metropolis.
- Noteboom, B. (2002): *Trust. Forms, Foundations, Functions, Failures and Figures*. Cheltenham: Edward Elgar.
- O'Mahony, S. (2003a): *Guarding the commons: How community managed software projects protect their work*. In: *Research Policy* 32, no. 7 (July 2003): 1179-1198.
- O'Mahony, S. (2003b): *Non-Profit Foundations and their Role in Community-Firm Software Collaboration*. In: *Making Sense of the Bazaar: Perspectives on Open Source and Free Software*. Boston: O'Reilly & Associates Publications.
- Oertel, B.; Joest, E.; Richter, M.; Rosdale, R.; Scheermesser, M.; Wölk, M. (2003): *Selbständig im Netz*. Werkstattberichte / IZT, Institut für Zukunftsstudien und Technologiebewertung; Nr. 58, Berlin, Juli 2003.
- Oertel, B.; Wölk, M.; Kreibich, R. (1998): *Neue Selbständigkeit im Netz*. In: Deutscher Bundestag (Hrsg.): *Arbeitswelt in Bewegung. Trends, Herausforderungen, Perspektiven*. Enquete-Kommission Zukunft der Medien in Wirtschaft und Gesellschaft. Deutschlands Weg in die Informationsgesellschaft, ZV Zeitungs-Verlag, Bonn.
- Oliver, P.; Marwell, G. (1992): *Mobilizing technologies for collective action*. In: Morris, A.; Mueller, C. (eds.): *Frontiers in social movement theory*. Binghamton, N.Y.: Vail-Ballou Press, 251-272.
- Orlikowski, W. (1992): *The Duality of Technology. Rethinking the concept of technology in Organizations*. In: *Organization Science*, Vol. 3, No. 3, Aug. 1992, 398-427
- Osterloh, M. (2002): *Open Source Software Production: The Magic Cauldron?* Working Paper of the University of Zürich, Prepared for the LINK Conference 2002 in Copenhagen.

- Osterloh, M.; Weibel, A. (2000): *Ressourcensteuerung in Netzwerken: Eine Tragödie der Allmende?* In: Sydow, J.; Windeler, A. (Hrsg.): *Steuerung von Netzwerken*, Opladen: Westdeutscher Verlag, 88-106.
- Ostrom, E. (1990): *Governing the Commons*. New York: Cambridge UP.
- Oughton, C.; Whittam, G. (1997): *Competition and cooperation in the small firm sector*. In: *Scottish Journal of Political Economy*, Vol. 44, No. 1, February 1997.
- Pal, N.; Madanmohan, T.R. (2002): *Competing on Open Source: Strategies and Practise*. Working Paper. <http://opensource.mit.edu/papers/madanmohan.pdf>, 20.04.2007.
- Pappi, F. U.; König, T. (1995): *Informationsaustausch in politischen Netzwerken*. In: Jansen, D.; Schubert, K. (Hrsg.): *Netzwerke und Politikproduktion. Konzepte, Methoden, Perspektiven*. Marburg: Schüren, 111-131.
- Paris, R. (1998): *Stachel und Speer. Machtstudien*. Frankfurt/Main: Suhkamp.
- Parsons, T. (1949): *The motivation of economic activities*. In: *Essays in sociological theory, pure and applied*. New York: The Free Press, 200-217.
- Parsons, T. (1972): *Das System moderner Gesellschaften*. München: Juventa Verlag.
- Parsons, T. (1976): *Zur Theorie sozialer Systeme*. Opladen: Westdeutscher Verlag.
- Pfaffmann, O.; Hornschild, K. (1999): *Neuere Erkenntnisse zur Bedeutung von FuE-Kooperationen industrieller Unternehmen. Ein Literaturüberblick*. In: *Konjunkturpolitik*, 45 Jg. H. 1.
- Picot, A.; Dietl, H.; Franck, E. (2002): *Organisation. Eine ökonomische Perspektive*. Stuttgart: Schäffer-Poeschl, 3. überarb. Aufl.
- Picot, A.; Reichwald, R.; Wigand, R. (1996): *Die grenzenlose Unternehmung. Information, Organisation und Management. Lehrbuch zur Unternehmensführung im Informationszeitalter*. Wiesbaden: Gabler.
- Pies, I. (2001): *Transaktion versus Interaktion, Spezifität versus Brisanz und die raison d'être korporativer Akteure – Zur konzeptionellen Neuausrichtung der Organisationsökonomik*. In: Pies, I.; Leschke, M. (Hrsg.): *Oliver Williamsons Organisationsökonomik*. Tübingen: 95-119.
- Pinch, T.; Bijker, W. (1987): *The social construction of facts and artefacts: Or how the sociology of science and the sociology of technology might benefit each other*. In: Bijker, W.; Hughes, T.; Pinch, T.: (ed.): *The social construction of technological systems. New directions in the Sociology and History of Technology*. 3<sup>rd</sup> Printing, Cambridge, Mass.: MIT Press, 17-51.
- Pirker, R. (2000): *Die Unternehmung als soziale Institution. Eine Kritik der Transaktionskostenerklärung der Firma*. In: Ortman; G.; Sydow, J.; Türk, K. (Hrsg.): *Theorien der Organisation. Die Rückkehr der Gesellschaft*. Wiesbaden: Westdeutscher Verlag, 2. Aufl., 67-80.
- Plehwe, D. (2000): *Neue Multis als transnationale Vernetzungsunternehmen*. In: Dörrenbacher, C.; Plehwe, D. (Hrsg.): *Grenzenlose Kontrolle? Organisatorischer Wandel und politische Macht multinationaler Unternehmen*, Berlin.
- Pohlmann, M. (1996): *Industrielle Netzwerke. Antagonistische Kooperationen an der Schnittstelle Beschaffung-Zulieferung*. München, Mering: Hampp.
- Popitz, H. (1986): *Phänomene der Macht. Autorität-Herrschaft-Gewalt-Technik*. Tübingen: Siebeck.
- Poulsen M.-B. (2001): *Competition and cooperation: what roles in scientific dynamics?* *International Journal of Technology Management*, Vol. 22, No.7/8, pp. 782-793.
- Powell, W. (1991): *Expanding the scope of institutional analysis*. In: Powell, W.; DiMaggio, P. (Eds.): *The New Institutionalism in Organizational Analysis*. Chicago: University of Chicago Press, 183-203.
- Powell, W. (1996): *Weder Markt noch Hierarchie: Netzwerkartige Organisationsformen*. In: Kenis, P.; Schneider, V. (Hrsg.): *Organisation und Netzwerk. Institutionelle Steuerung in Wirtschaft und Politik*. Frankfurt/Main: Campus Verlag, 213-272.
- Powell, W., Smith-Doerr, L. (1994): *Networks and economic life*. In: Smelser, N., Swedberg, R. (Hg.): *The handbook of economic sociology*. Princeton: Princeton university press, 368-402.
- Rammert, W. (1997): *Innovation im Netz. Neue Zeiten für technische Innovationen: heterogen verteilt und interaktiv vernetzt*. In: *Soziale Welt*, Jg. 48, 397-416.
- Rammert, W. (2000): *Technik aus soziologischer Perspektive*. Opladen: Westdeutscher Verlag.
- Rauch, H. (1983): *Partizipation und Leistung in Großgruppen-Sitzungen. Qualitative und quantitative Vergleichsanalyse von 20 Fallstudien zum Sitzungsprozeß entscheidungsfindender Großgruppen*. In: *Kölner Zeitschrift für Soziologie*, 35, 256-275.

- Raven, B. H. (1992): *A power/interaction model of interpersonal influence: French and Raven thirty years later*. In: *Journal of Social Behavior and Personality*, 7, 217-244.
- Raymond, E. (1998a): *The cathedral and the bazaar*. In: *First Monday*, Vol. 3, Nr. 3, [http://firstmonday.org/issues/issue3\\_3/raymond/](http://firstmonday.org/issues/issue3_3/raymond/), 20.04.2007.
- Raymond, E. (1998b): *Homesteading the noosphere*. In: *First Monday*, Vol. 3, Nr. 10, [http://firstmonday.org/issues/issue3\\_10/raymond/](http://firstmonday.org/issues/issue3_10/raymond/), 20.04.2007.
- Raymond, E. (1999a): *The magic cauldron*, <http://www.catb.org/~esr/writings/magic-cauldron/>, 20.04.2007.
- Raymond, E. (1999b): *A Brief History of Hackerdom*. In: DiBona, C.; Ockman, S.; Stone, M. (Eds.): *F/OSSs: Voices from the F/OSS Revolution*. Sebastopol, CA: O'Reilly & Associates.
- Reichwald, R.; Baethge, M.; Brakel, O.; Cramer, J.; Fischer, B.; Paul, G. (2004): *Die neue Welt der Mikrounternehmen*. Wiesbaden: Gabler.
- Reichwald, R.; Bieberbach, F.; Hermann, M. (1999): *SOHO (Small Office/Home Office) – Innovative Arbeitsformen im Dienstleistungsbereich*. In: *MittAB*, 32. Jg., Heft 4, 461-473.
- Reifer, D. J. (2000): *Web Development. Estimating Quick-to-Market Software*, *IEEE Software*, November/December 2000, 57-64.
- Reis, C.; Fortes, R. (2002): *An overview of the software engineering process and tools in the mozilla project*. <http://opensource.mit.edu/papers/reismozilla.pdf>; 20.04.2007.
- Reiß, M. (1998): *Mythos Netzwerkorganisation*. In: *Zfo*, 4/1998, 224-229.
- Renz, C. (2003): *Von der Anweisung zur Zielvereinbarung. Neue Formen der Steuerung und Koordinierung in der Industrie*. München und Mering: Hampp Verlag.
- Richter, R.; Furubotn, E. (1996): *Neue Institutionenökonomik. Eine Einführung und kritische Würdigung*. Tübingen: Mohr.
- Ripperger, T. (1998): *Ökonomik des Vertrauens. Analyse eines Organisationsprinzips*. Tübingen: Mohr.
- Robles, G. (2004): *(BETA3) CVS Analysis for the KDE project*. <http://libresoft.dat.escet.urjc.es/cvsanal/kde3-cvs/index.php?menu=Statistics>, 20.04.2007.
- Robles, G.; Scheider, H.; Tretkowski, I.; Weber, N. (2001): *Who is Doing It? A research on Libre Software developers*, <http://widi.berlios.de>, 20.04.2007.
- Rölle, D.; Blättel-Mink, B. (1998): *Netzwerke in der Organisationssoziologie – neuer Schlauch für alten Wein?* In: *ÖZS*, 23 Jg., 3/1998, 66-87.
- Ropohl, G. (2001): *Das neue Technikverständnis*. In: Ropohl, G. (Hrsg.): *Erträge der Interdisziplinären Technikforschung. Eine Bilanz nach 20 Jahren*. Berlin: Erich Schmidt Verlag. 11-30.
- Ropohl, G. (2004): *Was tun Ingenieure und was müssen sie dazu wissen? – Versuch einer Systematik*. In: Banse, G.; Ropohl, G. (Hrsg.): *Wissenskonzepte für die Ingenieurpraxis. Technikwissenschaften zwischen Erkennen und Gestalten*. VDI-Report 35, Düsseldorf: VDI.
- Rosenstiel, L. v.; Molt, W.; Rüttinger, B. (1995): *Organisationspsychologie*. Stuttgart: Kohlhammer, 8. Aufl.
- Rossi, M. (2004): *Decoding the „Free/Open Source (F/OSS) Software Puzzle“ – A survey of theoretical and empirical contributions*. Working Paper, Università di Siena, dipartimento di economia politica, N. 424, Aprile 2004, <http://opensource.mit.edu/papers/rossi.pdf>, 20.04.2007.
- Rucht, D.; Blattert, B.; Rink, D. (1997): *Soziale Bewegungen auf dem Weg zur Institutionalisierung? – Zum Strukturwandel „alternativer“ Gruppen in beiden Teilen Deutschlands*. Frankfurt/Main: Campus Verlag.
- Runkehl, J. (1998): *Sprache und Kommunikation im Internet: Überblick und Analysen*. Opladen: Westdeutscher Verlag.
- Sachverständigenrat Gesundheitswesen (2005): *Jahresgutachten 2005 – Koordination u. Qualität im Gesundheitswesen*. [http://www.sozialpolitik-aktuell.de/berichte\\_gesundheit.shtml](http://www.sozialpolitik-aktuell.de/berichte_gesundheit.shtml), 20.04.2007.
- Sader, M. (1991): *Psychologie der Gruppe*. München: Juventa, 2002, 8. Aufl.
- Sahlins, M. (2005): *Zur Soziologie des primitiven Tauschs*. In: Adloff, F.; Steffen M. (Hrsg.): *Vom Geben und Nehmen. Zur Soziologie der Reziprozität*. Frankfurt/Main: Campus Verlag, 73-94.
- Sartori, G. (1984): *Selbsterstörung der Demokratie? Mehrheitsentscheidungen und Entscheidungen in Gremien*. In: Guggenberger, B.; Offe, C. (Hrsg.): *An den Grenzen der*

- Mehrheitsdemokratie. Politik und Soziologie der Mehrheitsregel, Opladen: Westdeutscher Verlag.
- Sartori, G. (1992): *Demokratiethorie*. Darmstadt: Wissenschaftliche Buchgesellschaft.
- Sauer, D. (2001): *Unternehmensreorganisation und Entgrenzung von Arbeit – Thesen zum Umbruch*. In: Martens, H.; Peter, G.; Wolf, F. O. (Hrsg.): *Zwischen Selbstbestimmung und Selbstaussbeutung. Gesellschaftlicher Umbruch und neue Arbeit*. Frankfurt/Main: Campus Verlag, 27-38.
- Saxenian, A. (1996): *Regional Advantage: Culture and Competition in Silicon Valley and Route 128*. Harvard University Press, Cambridge, MA.
- Scacchi, W. (2002a): *Is Open Source Software Development Faster, Better, and Cheaper than Software Engineering?* In: 2nd. ICSE Workshop on Open Source Software Engineering, May, 2002, Orlando, FL.
- Scacchi, W. (2002b): *Understanding the Requirements for Developing Open Source Software Systems*. In: IEE Proceedings – Software, Vol. 149, No. 1, Febr. 2002.
- Scacchi, W.; Elliott, M. (2004): *Mobilization of Software Developers: The Free Software Movement*. Working Paper, August 2004, Revised version to appear in *Information, Technology, and People*, Festschrift for Rob Kling.
- Schäfers, B. (1999): *Entwicklung der Gruppensoziologie und Eigenständigkeit der Gruppe als Sozialgebilde*. In: Schäfers, B. (Hrsg.): *Einführung in die Gruppensoziologie*. Wiesbaden: Quelle & Meyer, 3. Aufl.
- Scharpf, F. W. (1992a): *Einführung: Zur Theorie von Verhandlungssystemen*. In: Arthur B. et al. (Hrsg.), *Horizontale Politikverflechtung: Zur Theorie von Verhandlungssystemen*. Frankfurt/Main: Campus Verlag, 11-27.
- Scharpf, F. W. (1992b): *Koordination durch Verhandlungssysteme: Analytische Konzepte und institutionelle Lösungen*. In: Arthur Benz et al. (Hrsg.), *Horizontale Politikverflechtung: Zur Theorie von Verhandlungssystemen*. Frankfurt/Main: Campus Verlag, 51-96.
- Schattenhofer, K. (1992): *Selbstorganisation und Gruppe. Entwicklungs- und Steuerungsprozesse in Gruppen*. Opladen: Westdeutscher Verlag.
- Schattenhofer, K. (1997): *Selbstorganisation und Gruppe. Eine Untersuchung zur Selbststeuerung in selbstorganisierten Gruppen*. In: Langthaler, W.; Schiepek, G. (Hrsg.): *Selbstorganisation und Dynamik in Gruppen. Beiträge zu einer systemwissenschaftlich orientierten Psychologie der Gruppe*. Münster: LIT.
- Schimank, U. (2000): *Handeln und Strukturen. Einführung in die akteurstheoretische Soziologie*. Weinheim: Juventa Verlag.
- Schimank, U. (2005): *Organisationsgesellschaft*. In: Jäger, W.; Schimank, U. (Hrsg.): *Organisationsgesellschaft. Facetten und Perspektiven*. Wiesbaden: Verlag für Sozialwissenschaft, 19-50.
- Schmid, J. (1997): *Der Personal Computer (1974 – 1985). Architektonische Innovation und vertikale Desintegration*. In: Weyer, et. al.: *Technik, die Gesellschaft schafft. Soziale Innovation als Ort der Technikgenese*. Berlin: 147–226.
- Schmidt, H. (1991): *Philosophisches Wörterbuch*. Neu bearb. von Georgi Schischkoff, 22. Aufl., Stuttgart: Kröner, Stichwort Arbeit.
- Schmiede, R. (2003): *Informationstechnik im gegenwärtigen Kapitalismus*. In: Böhme, G.; Manzei, A. (Hrsg.): *Kritische Theorie der Technik und der Natur*, München: Wilhelm Fink Verlag, pp. 173-183.
- Schmiede, R. (2006): *Wissen und Arbeit im „Informational Capitalism“*. In: Baukrowitz, A.; Berker, T.; Boes, A.; Pfeiffer, S.; Schmiede, R.; Will, M. (Hrsg.): *Informatisierung der Arbeit ? Gesellschaft im Umbruch*. Berlin: edition sigma.
- Schubert, H.; Fürst, D.; Rudolph, A. (2001): *Regionale Akteursnetzwerke*. Opladen: Leske und Budrich Verlag.
- Schüle, J. (1987): *Theorie der Institution. Eine dogmengeschichtliche und konzeptionelle Analyse*. Opladen: Westdeutscher Verlag.
- Schulz, A.; Schmid, A.; Krömmelbein, S. (1999): *Telearbeit – durch eine neue Arbeitsform zu mehr Beschäftigung?* In: WSI Mitteilungen, 10/1999, 711-722.
- Schumacher, U. (2003): *Lohn und Sinn. Individuelle Kombinationen von Erwerbsarbeit und freiwilligem Engagement*. Opladen: Leske und Budrich Verlag.

- Scott, R. (1991): *Unpacking Institutional Arguments*. In: Powell, W.; DiMaggio, P. (Eds.): The new institutionalism in organizational analysis. Chicago: Univ. of Chicago Press, 164-182.
- Scott, R.; Meyer, J. (1991): *The organization of societal sectors: Propositions and early evidence*. In: Powell, W.; DiMaggio, P. (Eds.): The New Institutionalism in Organizational Analysis. Chicago: Univ. of Chicago Press, 108-140, überarbeitete Version des Beitrags: The organization of societal sectors. In: Scott, R.; Meyer, J. (Eds.): Organizational environments: Ritual and rationality. Beverly Hills, 129-153.
- Scott, W. (2000): *Institutions and Organizations*. Thousand Oaks, CA: Sage Publications, 2. ed., 1995.
- Semlinger, K. (2003): *Vertrauen als Kooperationshemmnis- Kooperationsprobleme von kleinen und mittleren Unternehmen und Auswege aus der Vertrauensfalle*. In: Hirsch-Kreinsen, H.; Wannöffel, M. (Hrsg.): Netzwerke kleiner Unternehmen. Praktiken und Besonderheiten internationaler Zusammenarbeit. Berlin: Ed. Sigma, 61-88.
- Senge, K. (2005): *Der Neo-Institutionalismus als Kritik der ökonomistischen Perspektive*. Dissertation. <http://elib.tu-darmstadt.de/diss/000620/>, 20.04.2007.
- Serries (2005): *Die Bedeutung der intrinsischen Motivation in Prinzipal-Agent-Beziehungen am Beispiel der Beratungsstellen kirchlicher Wohlfahrtsverbände*. Dissertation, Wirtschaftswissenschaftliche Fakultät, Universität Göttingen.
- Setzen, K. (1999): *Gruppen im Sport*. In: Schäfers, B. (Hrsg.): Einführung in die Gruppensoziologie. Wiesbaden: Quelle & Meyer, 3. Aufl., 269-286.
- Shah, S. (2000): *Sources and Patterns of Innovation in a Consumer Products Field: Innovations in Sporting Equipment*. Working Paper of the Massachusetts Institute of Technology, Sloan Working Paper #4105 March 2000.
- Shapiro, C.; Varian, H. (1998): *Information Rules: a Strategic Guide to the Network Economy*. 1. Auflage, Boston, USA: Harvard Business School Press.
- Simmel, G. (1903): *Soziologie der Konkurrenz*. In: Neue Deutsche Rundschau (Freie Bühne), 14. Jg., Heft 10, Okt. 1903, 1009-1023 oder <http://socio.ch/sim/kon03.htm>, 20.04.2007.
- Simmel, G. (1958): *Soziologie. Untersuchungen über die Formen der Vergesellschaftung*. Berlin: Dunker & Humblot, 4. Aufl.
- Simon, H. (1991): *Organizations and Markets*. In: Journal of Economic Perspectives. Vol. 5, No. 2, 25-44.
- Söderberg, J. (2002): *Copyleft vs. Copyright: A Marxist critique*. In: First Monday, Vol. 7, Nr. 3, [http://firstmonday.org/issues/issue7\\_3/soderberg/](http://firstmonday.org/issues/issue7_3/soderberg/), 20.04.2007.
- Sofsky, W. (1983): *Die Ordnung sozialer Situationen. Theoretische Studien über die Methoden und Strukturen sozialer Erfahrung und Interaktion*. Opladen: Westdeutscher Verlag.
- Sommerville, I. (2001): *Software Engineering*. Harlow, England: Addison-Wesley, 6. Ed.
- Spieß, E. (1998): *Das Konzept der Empathie*. In: Spieß, E. (Hrsg.): Formen der Kooperation – Bedingungen und Perspektiven. Göttingen: Hogrefe, 53-62.
- Staber, U. (2000): *Steuerung von Unternehmensnetzwerken: Organisationstheoretische Perspektiven und soziale Mechanismen*. In: Sydow, J.; Windeler, A. (Hrsg.): Steuerung von Netzwerken, Opladen: Westdeutscher Verlag, 58-87.
- Stegbauer, C. (2002): *Reziprozität. Einführung in soziale Formen der Gegenseitigkeit*. Wiesbaden: Westdeutscher Verlag.
- Steger, R. (2003): *Netzwerkentwicklung im professionellen Bereich dargestellt am Modellprojekt REGINE und dem Beraternetzwerk zetTeam*. Materialien aus dem Institut für empirische Soziologie an der Friedrich-Alexander-Universität Erlangen-Nürnberg, 6/2003, Nürnberg: IfeS.
- Stichweh, R. (1994): *Wissenschaft, Universität, Professionen: soziologische Analysen*. Frankfurt/Main: Suhrkamp Verlag
- Streeck, W.; Schmitter, P. C. (1985): *Community, Market, State – and Associations? The Prospective Contribution of Interest Governance to Social Order*. In: European Sociological Review 1 (2), 119-138.
- Sydow, J. (1992): *Strategische Netzwerke: Evolution und Organisation*. Wiesbaden: Gabler.
- Sydow, J.; Windeler, A. (2000): *Steuerung von und in Netzwerken – Perspektiven, Konzepte, vor allem aber offene Fragen*. In: Sydow, J.; Windeler, A. (Hrsg.): Steuerung von Netzwerken, Opladen: Westdeutscher Verlag, 1-24.

- Sydow, J.; Windeler, A. (2004): *Projektnetzwerke: Management von (mehr als) temporären Systemen*. In: Sydow, J.; Windeler, A. (Hrsg.): *Organisation der Content-Produktion*. Wiesbaden: Verlag für Sozialwissenschaft, 37-54.
- Taubert, N. (2006): *Produktive Anarchie? Netzwerke freier Softwareentwicklung*. Bielefeld: Transcript.
- Taylor, R. (2004): *Extending conceptual boundaries: work, voluntary work and employment*. In: *Work, employment and society*. Vol. 18 (1): 29-49.
- Teubner, G. (1996): *Die vielköpfige Hydra: Netzwerke als kollektive Akteure höherer Ordnung*. In: Kenis, P.; Schneider, V. (Hrsg.): *Organisation und Netzwerk. Institutionelle Steuerung in Wirtschaft und Politik*. Frankfurt/Main: Campus Verlag, 535-562.
- Theuvsen, L. (1999): *Transaktionskostentheorie: Anwendungen auf Non-Profit-Organisationen*. In: Edeling, T.; Jann, W.; Wagner, D. (Hrsg.): *Institutionenökonomie und Neuer Institutionalismus. Überlegungen zur Organisationstheorie*. Opladen: Leske und Budrich Verlag.
- Thiedecke, U. (2000a): *Einleitung*. In: Thiedecke, U. (Hrsg.): *Virtuelle Gruppen. Charakteristika und Problemdimensionen*. Wiesbaden: Springer, 7-22.
- Thiedecke, U. (2000b): *Virtuelle Gruppen. Begriff und Charakteristik*. In: Thiedecke, U. (Hrsg.): *Virtuelle Gruppen. Charakteristika und Problemdimensionen*. Wiesbaden: Springer.
- Thompson, D. (1998; Hrsg.): *The concise Oxford dictionary of current English*. Oxford: Oxford Unity Press, 9<sup>th</sup> ed.
- Thompson, M.; Ellis, R.; Wildavsky, A. (1990): *Cultural theory*. Boulder: Westview.
- Tichy, N.; Fombrun, C. (1979): *Network Analysis in Organizational Settings*. In: *Human Relations*, Nr. 11, 1979.
- Törpel, B.; Pipek, V.; Rittenbruch, M. (2002): *Evolving Use of Groupware in a Service Network*. In: *Int. Journal on CSCW, Special Issue on Evolving Use of Groupware*, Kluwer.
- Trautwein-Kalms, G. (1997): *Informationsgesellschaft und Arbeitswelt: Nur Technik, Markt, Deregulierung? – Zur Vision von virtuellen Unternehmen und neuer Selbständigkeit*. In: *WSI-Mitteilungen*, 50. Jg., Heft 3, 169-177.
- Trezzini, B. (1998): *Theoretische Aspekte der sozialwissenschaftlichen Netzwerkanalyse*. In: *Schweizerische Zeitschrift für Soziologie* 24, H. 3, 511-544.
- Trice, H.; Beyer, J. (1993): *The cultures of work organizations*. Englewood Cliffs, N.J.: Prentice Hall.
- Tuomi, I. (2001): *Internet, Innovation, and Open Source: Actors in the network*. In: *First Monday*, Vol. 6, Nr. 1, [http://firstmonday.org/issues/issue6\\_1/tuomi/](http://firstmonday.org/issues/issue6_1/tuomi/), 20.04.2007.
- Tuomi, I. (2002): *Networks of Innovation. Change and Meaning in the Age of the Internet*. Oxford: Oxford Univ. Press.
- Türk, K. (2000): *Organisation als Institution der kapitalistischen Gesellschaftsformation*. In: Ortmann, G.; Sydow, J.; Türk, K. (Hrsg.): *Theorien der Organisation. Die Rückkehr der Gesellschaft*. Wiesbaden: Westdeutscher Verlag, 2. Aufl., 124-176.
- Uzzi, B. (1997): *Social structure and competition in interfirm networks: The paradox of embeddedness*. In: *Administrative Science Quarterly*, 42, 36-67.
- Välimäki, M. (2003): *Dual Licensing in Open Source Software Industry*. In: *Systemes d'Information et Management* 1/2003, und <http://opensource.mit.edu/papers/valimaki.pdf>, 20.04.2007.
- Voß, G.; Rieder, K. (2005): *Der arbeitende Kunde. Wenn Konsumenten zu unbezahlten Mitarbeitern werden*. Frankfurt/Main: Campus Verlag.
- Voswinkel, S. (2005): *Die Organisation der Vermarktlichung von Organisationen – Das Beispiel erfolgsbezogenen Entgelts*. In: Jäger, W.; Schimank, U. (Hrsg.): *Organisationsgesellschaft. Facetten und Perspektiven*. Wiesbaden: Verlag für Sozialwissenschaft, 287-312.
- Walgenbach, P. (1999): *Institutionalistische Ansätze in der Organisationstheorie*. In: Kieser A. (Hrsg.): *Organisationstheorien*. Stuttgart, Berlin, Köln: Kohlhammer, 3.Aufl., 319-354.
- Walgenbach, P. (2002): *Neoinstitutionalistische Organisationstheorie – State of the Art und Entwicklungslinien*. In: Schreyögg, G.; Peter, C. (Hrsg.): *Theorien des Managements*. Wiesbaden: Gabler, 155-202.
- Waters, M. (1989): *Collegiality, Bureaucratization, and Professionalization: A Weberian Analysis*. In: *American Journal of Sociology*, Vol. 94, No. 5, March 1989, 945-972

- Waterson, P. E.; Clegg, C. W.; Axtell, C. M. (1997): *The dynamics of work organization, knowledge and technology during software development*. In: Int. J. Human-Computer Studies (1997) 46, 79 – 101.
- Weber, K. (2004): *Philosophische Grundlagen und mögliche Entwicklungen der Open-Source- und Free-Software-Bewegung*. In: Gehring, R. A.; Lutterbeck, B. (Hrsg.): Open Source Jahrbuch 2004. Zwischen Softwareentwicklung und Gesellschaftsmodell. Berlin: Lehmanns Media.
- Weber, M. (1980): *Wirtschaft und Gesellschaft*. Tübingen: Mohr, 1921, 5. rev. Aufl.
- Weber, S. (2000): *The political economy of open source software*. (Berkley Roundtable on the International Economy) BRIE Working Paper, 140 (June 2000), <http://opensource.mit.edu/online-papers.php>, 20.04.2007.
- Weber, S. (2004). *The Success of Open Source*. Cambridge, MA: Harvard UP.
- Wegberg, M. v.; Berends, P. (2000): *Competing communities of users and developers for computer software: Competition between open source software and commercial software*. NIBOR Working Paper, University of Maastricht, Faculty of Economics and Business Administration, NIBOR/RM/00/001. <http://edata.ub.unimaas.nl/www-edocs/loader/file.asp?id=283>, 20.04.2007.
- Weigle, J.; Krcmar, H. (2000): *Zur Funktionsweise Virtualisierter Unternehmen. Informations- und Kommunikationstechnologie in einem Projekt der Rauser Advertainment AG*. Arbeitsbericht Nr. 161, Universität Hohenheim, Institut für Betriebswirtschaftslehre, Lehrstuhl für Wirtschaftsinformatik.
- Wehrich, M.; Dunkel, W. (2003): *Abstimmungsprobleme in Dienstleistungsbeziehungen. Ein handlungstheoretischer Zugang*. In: Kölner Zeitschrift für Soziologie und Sozialpsychologie, Jg. 55, H.4, 2003, 758-781.
- Weise, P. (2000): *Kooperation und unternehmungsinterne Arbeitsbeziehungen*, in: Krömmelbein, S.; Schmid, A. (Hrsg.): Globalisierung, Vernetzung und Erwerbsarbeit. Theoretische Zugänge und empirische Entwicklungen, Wiesbaden, 179-202.
- Wellman, B. (2000): *Die elektronische Gruppe als soziales Netzwerk*. In: Thiedecke, U. (Hrsg.): Virtuelle Gruppen. Charakteristika und Problemdimensionen. Wiesbaden: Springer, 134-167.
- Welz, F.; Ortmann, R. (1992): *Das Softwareprojekt. Projektmanagement in der Praxis*. Frankfurt/Main; New York: Campus Verlag.
- Wenger, E. (1998): *Communities-of-Practice. Learning, Meaning and Identity*. Cambridge: Cambridge University Press.
- Wenny, A. (1999): *Ein-Personen-Unternehmen. Die Rolle der Unternehmensform im wirtschaftlichen Wandel, die soziale und wirtschaftliche Stellung von Selbständigen ohne eigene Beschäftigte*. Marburg: Tectum.
- Wetzel, J. (1998): *Problemlösen in Gruppen: Miteinander ist besser als gegeneinander*. In: Ardelt-Gattinger, E.; Lechner, H.; Schlögl, W. (Hrsg.), Gruppendynamik. Anspruch und Wirklichkeit der Arbeit in Gruppen, Göttingen: Verlag für Angewandte Psychologie, 113-127.
- Weyer, J. (2000): *Einleitung. Zum Stand der Netzwerkforschung in den Sozialwissenschaften*. In: Weyer, J. (Hg.): Soziale Netzwerke. München u.a.: Oldenbourg, 1-34.
- Wiesenthal, H. (2005): *Markt, Organisation und Gemeinschaft als 'zweitbeste' Verfahren sozialer Koordination*. In: Jäger, W.; Schimank, U. (Hrsg.): Organisationsgesellschaft. Facetten und Perspektiven. Wiesbaden: Verlag für Sozialwissenschaft, 223-264.
- Wildavsky, A. (1986): *On Collaboration*. In: PS, Vol. 19, No. 2, Spring 1986, 237-248.
- Wilke, H.; van Knippenberg, A. (1992): *Gruppenleistung*. In: Stroebe, W.; Hewstone, M.; Codol, J.-P.; Stephenson, G. (Hrsg.): *Sozialpsychologie. Eine Einführung*. Berlin: Springer. 2. Aufl.
- Williams, S. (2002): *Free as in Freedom – Richard Stallman's Crusade for Free Software*. <http://www.oreilly.com/openbook/freedom/>, 20.04.2007.
- Williamson, O. (1985): *The economic institution of capitalism*. New York: Free Press.
- Williamson, O. (1991): *Comparative economic organization: the analysis of discrete structural alternatives*. In: Administrative Science Quarterly, 36, 269-296.
- Williamson, O. (1993): *Calculativeness, trust, and economic organization*. In: Journal of Law and Economics, Vol. 36, April 1993, 453-486.
- Willke, G. (1998): *Die Zukunft der Arbeit*. Hannover: Niedersächsische Landeszentrale für politische Bildung.

- Willke, H. (1987): *Institution*. In: Görlitz, A.; Prätorius, R. (Hrsg.): Handbuch Politikwissenschaft. Reinbeck: Rowohlt, 162-166.
- Windeler, A. (2001): *Unternehmensnetzwerke. Konstitution und Strukturation*. Wiesbaden: Westdeutscher Verlag.
- Windeler, A. (2003): *Spuren im Netzwerkdschungel: Typen von Unternehmensnetzwerken und Besonderheiten ihrer Koordination*. In: Hirsch-Kreinsen, H.; Wannöffel, M. (Hrsg.): Netzwerke kleiner Unternehmen. Praktiken und Besonderheiten internationaler Zusammenarbeit. Berlin: Ed. Sigma, 35-60.
- Windeler, A.; Lutz, A.; Wirth, C. (2000): *Netzwerksteuerung durch Selektion – Die Produktion von Fernsehserien in Projektnetzwerken*. In: Sydow, J.; Windeler, A. (Hrsg.): Steuerung von Netzwerken, Opladen: Westdeutscher Verlag, 178-205.
- Wolf, H. (2000): *Das Netzwerk als Signatur der Epoche? Anmerkungen zu einigen Beiträgen zur soziologischen Gegenwartsanalyse*. In: Arbeit, Heft 2, Jg. 9, 95-104.
- Wolff, K. (1991): *Schwarzarbeit in der Bundesrepublik Deutschland: eine mikroanalytische Untersuchung*. Frankfurt/Main;New York: Campus Verlag.
- Working Group on Libre Software (2000): *Open Source: Information society opportunities for Europe?* Version 1.2. <http://eu.conecta.it/paper.pdf>, 20.04.2007.
- Yamauchi, Y.; Yokozawa, M.; Shinohara, T.; Ishida, T. (2000): *Collaboration with Lean Media: How Open-Source Software Succeeds*. In: ACM Conference on Computer Supported Cooperative Work (CSCW2000), Philadelphia, PA, December 2000.
- Zeitlyn, D. (2003): *Gift economies in the development of open source software: Anthropological reflections*. In: Research Policy, Special Issue on Open Source Software Development, Volume 32, Issue 7, 1287-1291 (July 2003)
- Zenger, T. (2002): *Crafting internal hybrids: Complementarities, common change initiatives, and the team-based organization*. In: International Journal of the Economics of Business, Vol. 9, No. 1, 2002, pp. 79-95.
- Zimmermann, T. (2004): *Open Source und Freie Software – soziale Bewegung im virtuellen Raum?* In: Gehring, R. A.; Lutterbeck, B. (Hrsg.): Open Source Jahrbuch 2004. Zwischen Softwareentwicklung und Gesellschaftsmodell. Berlin: Lehmanns Media.
- Zucker, L. (1986): *Production of trust: Institutional sources of economic structure, 1840-1920*. In: Research in Organizational Behavior 8, 53-111.
- Zucker, L. (1991): *The role of institutionalization in cultural persistence*. In: Powell, W.; DiMaggio, P. (Hrsg.): The new institutionalism in organizational analysis. Chicago: Univ. of Chicago Press, erstmals 1977 in American Sociological Review, Vol. 42, 726-743.
- Zuckerman, E. (2003, Eds.): *On Networks and markets by Rauch and Casella*. In: Journal of Economic Literature, Vol. XLI (June 2003), pp. 545-565.

## 9. Anhang

### Anhang A: Tabellen und Abbildungen

Anhang A Tab. A 1: Anzahl der Befragten nach Nationalität der Kommunikationspartner und der Personen der Selbstaussagen nach eigener Nationalität, absolut		
	Befragung	Selbstaussagen
	Mit KDElern aus welchem Land kommunizieren Sie regelmäßig?	Eigene Nationalität
Deutschland	53	19
Österreich, Schweiz	27	3
Frankreich, Italien, Spanien	23	5
Großbritannien, Irland	22	5
Osteuropa	9	4
Skandinavien	18	0
restliches Europa	14	7
USA, Kanada	31	6
restliches Amerika	5	0
Australien, Neuseeland	8	1
restliche Welt (z.B.: Asien, Afrika, etc.)	5	10
Weiß nicht	5	0
<b>Gesamt</b>	<b>55</b>	<b>60</b>

Erhebung: 55 beantwortete Fragebögen einer quantitativen schriftlichen Befragung; 62 Selbstaussagen auf der KDE-Homepage; Stand Mitte 2003

Anhang A Tab. A 2: Anzahl der Befragten und der Personen der Selbstaussagen nach IT-Ausbildung/Studium, absolut		
Haben Sie IT-Ausbildung/Studium oder Verwandtschaftliches absolviert?	Befragung <sup>a</sup>	Selbstaussagen <sup>b</sup>
IT-Ausbildung bzw. IT-Studium, z.B. Informatik	28	15
IT-verwandte Ausbildung bzw. Studium, z.B.: Mathe, Ing.wiss, Naturwiss.	13	10
Sonstige Ausbildung/Studium	2	3
Keine	6	2
<b>Gesamt</b>	<b>51</b>	<b>28</b>

<sup>a</sup> Höchster Bildungsabschluss: 29 Hochschulabschluss, 20 Abitur und 3 Realschul- und 1 Hauptschulabschluss  
<sup>b</sup> von 62 Interviewten sind oder werden 47 Akademiker

Erhebung: 55 beantwortete Fragebogen einer quantitativen schriftlichen Befragung; 62 Selbstaussagen auf der KDE-Homepage; Stand Mitte 2003

Anhang A Tab. A 3: Anzahl der Befragten und der Personen der Selbstaussagen nach Geschlecht, absolut

Geschlecht	Männlich	Weiblich	Gesamt
Befragung	52	1	53
Selbstaussagen	58	4	58

Erhebung: 55 beantwortete Fragebogen einer quantitativen schriftlichen Befragung; 62 Selbstaussagen auf der KDE-Homepage; Stand Mitte 2003

Anhang A Tab. A 4: Anzahl der Befragten und der Personen der Selbstaussagen nach Geburtsjahrkategorie, 2003, absolut

Geburtsjahr	<1965	1965-1969	1970-1974	1975-1979	1980-1984	>1984	Gesamt
Befragung	2	5	12	21	11	2	53
Selbstaussagen	3	5	23	21	5	1	58

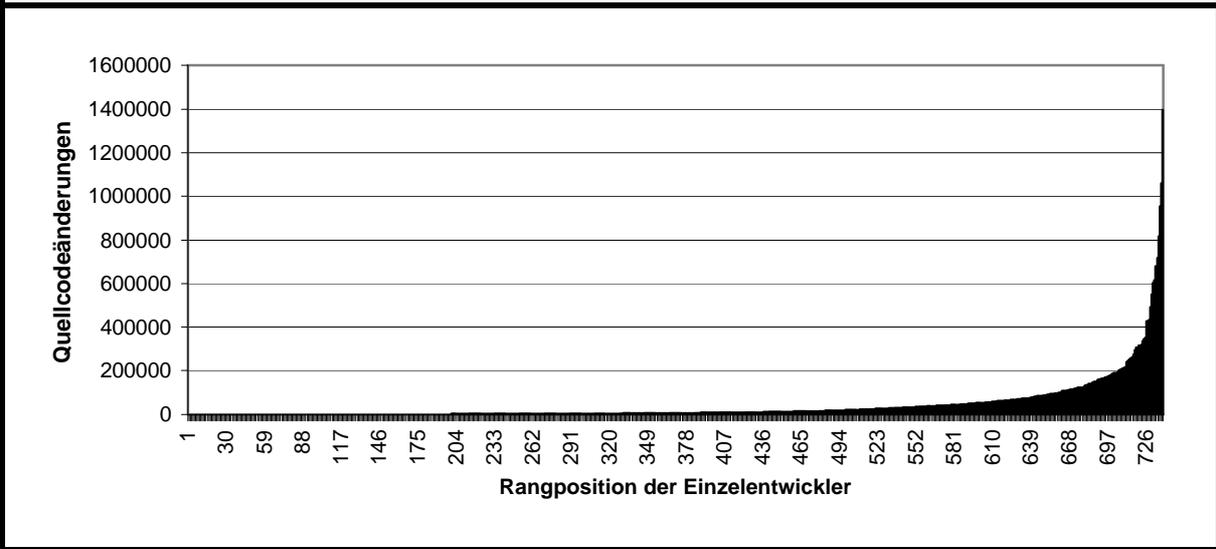
Erhebung: 55 beantwortete Fragebogen einer quantitativen schriftlichen Befragung; 62 Selbstaussagen auf der KDE-Homepage; Stand Mitte 2003

Anhang A Tab. A 5: Anzahl der Befragten und der Personen der Selbstaussagen nach Tätigkeiten, absolut

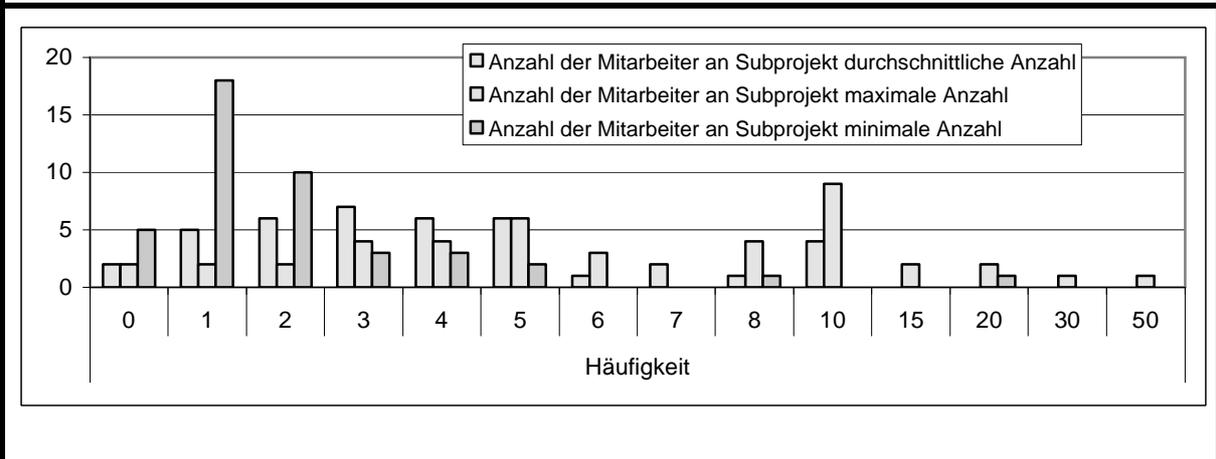
Tätigkeiten	Befragung <sup>a</sup>	Selbstaussagen <sup>b</sup>
Applikationen programmieren	40	15
Kernbibliotheken programmieren	19	
Dokumentieren	5	2
Übersetzen	4	7
Design/Grpfiken (Icons)	3	
Journalistisches Internetportal für das Projekt	7	
Fehlerberichte und Patches einsenden	19	1
Produkt nutzen/anwenden	23	
Betreuung einer Webseite	3	
Events/Messen organisieren	4	
Quellcode zu lauffähigem Maschinencode packen	1	1
Organisation des Open Source-Vereins	1	
Gesamt	55	26

<sup>a</sup> Mehrfachnennungen möglich; <sup>b</sup> Hauptsächliche Tätigkeit; Erhebung: 55 beantwortete Fragebogen einer quantitativen schriftlichen Befragung; 62 Selbstaussagen auf der KDE-Homepage; Stand Mitte 2003

Anhang A Abb. A 1: Verteilung der Mitarbeit bei KDE, Mitte 2003, 740 Entwickler<sup>263</sup>



Anhang A Abb. A 2: Häufigkeit der Nennung der Mitarbeiteranzahl an Subprojekten bei KDE (quant. Befragung), absolut



Erhebung: 55 beantwortete Fragebogen einer quantitativen schriftlichen Befragung; Stand Mitte 2003

<sup>263</sup> Die beiden letzten Rangpositionen 739 und 740 mit 3917579 und 8227683 Quellcodeänderungen wurden aus didaktischen Gründen bei der Grafik ausgelassen. Damit wird aber die Aussage der Grafik nicht verfälscht. Es wird G. Robles (2004) für das zur Verfügung stellen der Daten gedankt.

## **Anhang B: Interviewleitfaden für das Open-Source-Projekt**

Teil I: Die Struktur von Open-Source-Projekten

I-A) Produkt- und Projektbeschreibung

1. Was machen/produzieren Sie in Ihrem Team? (Produktbeschreibung)
  - 1.1 Was ist das hergestellte Produkt? Aus welchem Geschäftsfeld bzw. Branche kommt es? Nischenmarkt?
  - 1.2. Gibt es eine Produktdiversifizierung, d.h. verschiedene Produkte, die Sie herstellen?
  - 1.3. Gab es am Anfang einen großen Beitrag von einer Person bzw. von einem Unternehmen? (z.B. fertiges Programm mit Fehlern)
  - 1.4. Für wen wird das Produkt entwickelt? An wen wird da gedacht?
2. Wer ist an Ihrem Team beteiligt? (Projektbeschreibung) Wieviel sind beteiligt?
  - 2.1. Gibt es einen inneren Kreis und einen Dunstkreis?
  - 2.2. Wo wohnen und arbeiten die Netzwerkmitglieder?
  - 2.3. Welche Nationalität haben die Netzwerkteilnehmer? Was ist die Geschäftssprache?
  - 2.4. Welche Arbeitsaufgaben bzw. Tätigkeitsbereiche gibt es im Team?
3. Wie lange gibt es das Projekt schon?
  - 3.1. Wie ist Projekt bekannt geworden?
  - 3.2. Wie ist die Beziehung zu anderen Teams bzw. anderen Open-Source-Communities?
  - 3.3. Wie lange wird das Projekt noch zusammenbleiben?

I-B) Persönliches

4. Was ist Ihr persönlicher Status?
  - 4.1. Warum arbeiten Sie an diesem Projekt mit?
  - 4.2. Wieviel Arbeitszeit stecken Sie in das Projekt?
  - 4.3. Wann arbeiten Sie vor allem für das Projekt?
  - 4.4. Wo arbeiten Sie vor allem für das Projekt? (Arbeitsort)
  - 4.5. Arbeiten Sie an verschiedenen Projekten in Ihrem Projekt mit? Welche?
  - 4.6. Haben Sie schon Erfahrungen mit anderen Netzwerken gemacht? Nehmen Sie an verschiedenen Netzwerken teil?

I-C): Die Informationstechnik:

5. **Wie sieht Ihre technische Infrastruktur aus, damit Sie damit arbeiten können? Welche Technologien bzw. Werkzeuge benutzen Sie?**
  - 5.1. Hat das Projekt ein eigenes Internetportal?  
Wie viele nutzen diese? Welchen Service bietet das Portal an?  
(technische, rechtliche, ökonomische Beratung; Überwachung des Projektablaufs)  
Sind die Nutzer bzw. die Entwickler anonym?

Teil II: Die Funktionsweise von Arbeitsnetzwerken

Externe Allokation / Eintritt-Austritt:

6. **Wie kommt man in das Netzwerk bzw. Team rein und wie geht man wieder raus?**
  - 6.1. **Wie kam der Kontakt zum Netzwerk zustande?** Wie sind Sie auf das Netzwerk aufmerksam geworden? Woher haben Sie die Informationen bekommen? Ist das der typische Weg oder gibt es noch andere Wege?

**6.2. Wieviel Ein- und Austritte gibt es / Wie viele machen mit und verschwinden wieder?**

Gibt es ein ständiges Wechseln zwischen den verschiedenen OS-Projekten, wie Linux, KDE, Openoffice, etc. oder bleibt man bei einem?

Hat die Bekanntheit des Projekts Auswirkungen gehabt?

**6.3. Wer darf sich beteiligen? Darf sich jeder beteiligen?**

**Gibt es Regeln der Auswahl?**

Wie komme ich in den inneren und in den äußeren Kreis bzw. Dunstkreis rein?

Nach welchen Kriterien und wie wird ausgewählt?

Wie wird mit Neulingen umgegangen?

Haben sich die Kriterien verändert? Warum?

**6.4. Wer legt die Regeln fest? Wie sind die Einflußmöglichkeiten auf die Regeln?**

**6.5. Sind die Netzwerkmitglieder an das Projekt durch Verbindlichkeiten bzw.**

Abmachungen gebunden oder können sie jederzeit aussteigen? Warum ist das so?

Gibt es längerfristige Bindungen an ein Projekt?

Ist das Ende des Projekts festgelegt, ist ein Ende in Sicht oder wird es ewig weitergehen?

Leistungserstellung: Interne Verteilung der Arbeitsaufgaben

**7. Wie wird festgelegt wer was wann macht?**

**7.1. Wie werden die Arbeitsbeiträge im Netz erstellt?**

Was sind die Hauptaufgaben?

Sind die Arbeitsbeiträge groß oder eher klein?

Was wird lieber gesehen?

Wann werden die Arbeitsbeiträge abgeliefert?

Wie wird mit Überschneidungen bei den Arbeitsbeiträgen umgegangen? Wird PRG-Code wiederverwendet?

Wie lange dauert die Erstellung des Netzwerkproduktes?

**7.2. Wie wird die Arbeit aufgeteilt?**

Wer wählt die Arbeitsbeiträge aus?

Wer übernimmt welche Arbeitsaufgaben bzw. Tätigkeitsbereiche im Team und warum?

Wird der Arbeitsbeitrag vorher abgesprochen oder werden auch ungefragte Einsendungen verwendet?

Wer legt die Regeln fest? Wie sind die Einflußmöglichkeiten auf die Regeln?

Wie ist die Aufgabenverteilung, wenn neue Aufgaben anstehen, zu viel Arbeit da ist oder eine wichtige Person ausfällt?

Passiert dies häufig?

**7.3. Wer arbeitet wieviel? Wie ist der Arbeitsaufwand der Netzwerkmitglieder?**

Warum?

Wieviel Arbeitszeit wird in das Projekt gesteckt?

Wer legt den Arbeitsaufwand bzw. -zeit fest?

Wie hängt die Reputation eines Mitglieds mit dessen Arbeitsaufwand/Arbeitszeit zusammen?

Welcher Arbeitsaufwand wird als normal angesehen? Wird dies von allen so gesehen?

Wird auch mehr als dieser normale Arbeitsaufwand geleistet? (Arbeitsnorm)

Wann arbeiten die Netzwerkmitglieder an dem Projekt? Wo arbeiten die Netzwerkmitglieder für das Projekt?

Arbeiten die Netzwerkmitglieder an verschiedenen Projekten in diesem Projekt mit? Welche?

Haben die Netzwerkmitglieder schon Erfahrungen mit anderen Netzwerken gemacht?

Nehmen die Netzwerkmitglieder an verschiedenen Netzwerken teil?

- 7.4. Wie wird die Arbeit zusammengeführt/koordiniert?  
**Gibt es Regeln der Auswahl des Arbeitsbeitrags?** Wie erfolgt die Bewertung der Arbeitsbeiträge?  
Wer legt die Regeln fest? Wie sind die Einflußmöglichkeiten auf die Regeln?  
Gibt es einen Projektinitiator oder einen Teilbereichsverantwortlichen? Wofür ist er zuständig?  
Wie spielt bei der Bewertung des Beitrags die Reputation des Bewertenden eine Rolle?  
Haben sich die Kriterien zur Auswahl eines Beitrages verändert? Warum?
- 7.5. Wie wird mit dem Einbringen von Ressourcen umgegangen?  
Welche Ressourcen sind dies?  
Wie wird dies honoriert?
- 7.6. Gibt es eine gegenseitige Abhängigkeit im Projekt?  
Arbeiten Sie isoliert voneinander oder gibt es einen kontinuierlichen Austauschprozeß?  
Lernen Sie voneinander, wenn sie miteinander arbeiten? Warum? Was lernen sie?  
Ist man von der Arbeit eines anderen abhängig? Warum?  
Können Sie sich auf die Arbeit eines anderen Mitglieds verlassen?  
Wie werden Neulinge eingearbeitet bzw. eingewiesen?

Kontrolle:

**8. Gibt es so was wie eine Kontrolle? Welcher Art ist diese Kontrolle?**

**8.1. Wie wird die Qualität eines Beitrags bewertet?**

- Was sind die Kriterien der Bewertung?  
Ergibt sich dadurch eine höhere Qualität?  
Sind Netzwerkmitglieder durchschnittlich stärker motiviert? Deswegen bessere Qualität des Produkts?  
Werden nur die Produktbeiträge oder auch das Verhalten der Beteiligten bewertet?  
Muß der nächste Release besser als der letzte sein? Warum?  
**Spielt Vertrauen und Freundschaft bzw. freundschaftliche Gefühle eine Rolle?**  
Muß deswegen weniger kontrolliert werden?  
Wie sind die Einflußmöglichkeiten auf die Regeln?  
Wie spielt bei der Bewertung des Beitrags die Reputation des Bewertenden eine Rolle?  
Haben sich die Kriterien zur Auswahl eines Beitrages verändert? Warum?

**8.2. Wer darf bewerten?**

- Wer hat die Verantwortlichkeiten über welches Projekt?  
Wer legt die Verantwortlichkeiten fest?: Verteilung der Verantwortlichkeit Was darf er, was nicht? Warum? Welche Entscheidungsbefugnisse hat er und wo liegen deren Grenze? Wenn diese Personen etwas sagen, wird dies befolgt? Haben diese Macht?  
Wie ist die Führung?

Motivation/ Gratifikation/Belohnung:

**9. Welche Vorteile haben Sie von der Mitarbeit an dem Projekt gegenwärtig und künftig?**

- 9.1. Spielt die (zukünftige) Entlohnung eine Rolle? Warum?  
9.2. Oder spielen andere, (nicht-monetäre) Motive eine Rolle? Warum?  
9.3. Schwankt die Motivation? Warum steigt und sinkt die Motivation? Drückt Gruppenswang/-druck auf die eigene Motivation?

Netzwerkkultur

10. Identifizieren sie sich mit dem Projekt?  
Fühlen Sie sich dem Projekt verpflichtet?

- 10.1. Wem gehört das Produkt des Projekts? (**Eigentums- und Verfügungsrecht**)
- 10.2. Darf das Open-Source-Produkt verkauft / vermarktet werden? Wie spielt die Lizenz (BSD, LGPL, GPL) mit hinein? Ist es juristisch festgelegt?
- 10.3. Wenn Gewinne aus dem Open-Source-Produkt gemacht werden können, wer bekommt den Gewinn? (Gewinneinbehaltungsrechte)
- 10.4. Wer darf das Produkt verändern? Wer nutzen? (Nutzungsrechte; abh. von Lizenz)  
Ist der komplette/zeitliche Verkauf der Nutzungsrechte an Auftraggeber möglich?
- 10.5. Gab es Lizenzverstöße? Wurde schon gegen Lizenzverstöße geklagt?
- 10.6. Haben sich die Regeln der Nutzung und Vermarktung des gemeinsamen Produktes verändert? Warum? Wie sind die Einflußmöglichkeiten auf die Regeln?

Konflikte:

**12. Gibt es oder gab es Konflikte in der Community? Welcher Art sind diese Konflikte?**

**Was sind typische Konflikte?**

Wie oft kommen Konflikte vor? Was war bisher Ihr schlimmster Fall eines Konflikts?  
Was sind alltägliche Konflikte? Wie lange dauern Konflikte im Durchschnitt bzw. im Besonderen?

- 12.1. Wer ist in einem solchen Konflikt beteiligt? Gibt es verschiedene Konfliktparteien oder ist es nur ein Störer?

a) Verschiedene Konfliktparteien

- 12.2. Was sind die Konfliktthemen?

Wie ist der Ablauf des Konflikts? Gibt es Wortführer? Wie groß sind die Konfliktparteien?

b) Einzelner Störer

- 12.3. Wurde von Anfang an im Projekt kooperiert? Gibt es Leute, die nicht kooperieren?

Wie stören Sie? Kommt dies öfters vor? Bleiben diese Unkooperativen außen vor?  
Warum gehen Sie gegen solche Störer vor?

Was wird als Störung angesehen? Was ist noch geduldet?

Wer geht gegen Störer vor? Alle oder einer mit hoher Reputation?

Kann er etwas durchsetzen? Wird dies ohne weiteres anerkannt?

Gibt es Trittbrettfahrer? Wie erfährt man von Ihnen? Wie wird damit umgegangen?

**13. Wie wird mit dem Konflikt umgegangen?**

- 13.1. Wird der Konflikt untereinander ausgehandelt oder wird der Konflikt geschlichtet?

Hat die Stellung einer Konfliktpartei im Netzwerk Einfluß auf den Ablauf des Konflikts?

Wer setzt sich durch?

Über welches Medium erfolgt die Schlichtung bzw. Aushandlung?

Hat jemand die Letztentscheidungsmacht?

Wie schnell versöhnen Sie sich wieder und was reicht zur Versöhnung?

- 13.2. Was für Sanktionsmöglichkeiten stehen zur Verfügung?

Wie und wann werden die Sanktionsmöglichkeiten eingesetzt?

Wie oft verwenden Sie welche Strafen? Wird sofort bestraft?

Wie sind die Einflußmöglichkeiten auf die Anwendung der Sanktionsmöglichkeiten?

Haben sich die Kriterien zur Anwendung der Sanktionsmöglichkeiten verändert?

Warum?

- 13.3. Was passiert, wenn das Aushandeln bzw. die Konfliktschlichtung scheitert?  
(Grenzen des Netzwerks)

Haben Sie so was schon einmal erlebt?

Was ist der Grund für das Scheitern? Wo sehen Sie Probleme der Projekte, der Zusammenarbeit im Projekt? Warum funktionieren manche Projekte nicht? Starten nicht? Schlafen zwischendurch ein?

Der Einfluß von IT

**14. Welche Tools nutzen Sie wann für welchen Zweck? Welche sind die wichtigsten?**

- 14.1. Warum und Wofür nutzen Sie die verschiedenen IT-Werkzeuge? Wie erleichtert die IT den Eintritt und Austritt ins Projekt? Gibt es Stufen des Eintritts?  
Wie unterstützt IT die Leistungserstellung?  
Wie übernimmt die IT die Aufgabenverteilung im Projekt?  
Wie unterstützt die IT die Information über die Reputation der Mitglieder?  
Wie kontrolliert die IT die Arbeitsergebnisse? Wie unterstützt sie die Qualitätssicherung?
- 14.2. Wie wird Wissen weitergegeben bzw. gespeichert? Welche Rolle spielt dabei IT?  
Wie wird mit dem Wissen der verschiedenen Projektmitgliedern umgegangen? Wird das Wissen irgendwie dokumentiert? Oder wird das Wissen erhalten über wiederkehrende Zusammenarbeit?  
Gibt es eine Wissensdatenbank?  
Was passiert, wenn Wissen im Gesamtprojekt bzw. in einem Unterprojekt nicht vorhanden ist?
- 14.3. Wann nutzen Sie welche Technik?  
Wie oft nutzen Sie welche Technik bzw. Medien?  
Wenn Sie mit jemandem aus dem Projekt in Kontakt treten, welche Medien/Techniken nutzen Sie überwiegend? Wechseln Sie öfter das Medium bei gleichem Thema?
- 14.4. Sprechen Sie in Ihrem Projekt nur über das Projekt oder sprechen Sie auch über Privates, wie Ihre Hobbies, Freizeitaktivitäten, Interessensgebiete, Probleme, etc.?  
Welche Medien finden dann eine Anwendung?
- 14.5. Treffen Sie sich ab und zu mit anderen Projektmitgliedern?  
Wo oft nutzen Sie das persönliche Gespräch (face-to-face):  
Wann und wofür kommt es zu persönlichen Gesprächen?  
Wo treffen Sie sich/Haben sich getroffen?
- 14.6. Haben Sie oder Ihre Projektmitglieder Probleme mit der Technik?  
Was sind das für Probleme? Was wünschen Sie sich verändert?  
Wie gehen Sie mit der Masse an Informationen um?  
Wie stellen Sie die Authentizität einer Nachricht / Info / bugfix fest?



7. Wo würden Sie sich selbst einordnen im Gefüge von KDE? **Nur eine Angabe möglich**

- innerer Kreis/Kernbereich ..... 1
- äußerer Kreis (cvs-account-Zugang) ..... 2
- weiteres Umfeld (bug reports/patches einsenden) ..... 3
- Weiß nicht ..... 9

8. Welche Funktion haben oder hatten Sie in KDE? **Mehrfachnennungen möglich**

- Releasemaintainer ..... 1
- Vergabe von cvs-accounts/cvs-maintainer ..... 2
- Homepagepflege/Webmaintainer ..... 3
- Koordinator eines Subprojekts/eines Teams an KDElern ..... 4
- ständiger (Sub-)Projektteilnehmer ..... 5
- Aushilfe bei Subprojekten, bei denen es brennt/Springer ..... 6
- Werbung/Marketing für KDE ..... 7
- Mitglied beim KDE e.V. .... 8
- Vergabe von Mailinglisten-Accounts/Mailinglistenmaintainer (z.B. für core-devel) ..... 9
- Distributionsverpacker und -compilierer ..... 10
- Poweruser (häufiger Betatester und Bugreporter) ..... 11
- Unterstützungsnetzwerk für Releasemaintainer ..... 12
- Eine andere Tätigkeit ..... 13

und zwar: \_\_\_\_\_

9. Mit wie vielen KDElern kommunizieren Sie regelmäßig (persönlich, traditionelle Medien, elektronisch)? **Nur eine Angabe möglich**

- Keinem ..... 1
- 1-10 Personen ..... 2
- 11-20 Personen ..... 3
- 21-50 Personen ..... 4
- Mehr als 50 Personen ..... 5

10. Mit KDElern aus welchem Land kommunizieren Sie regelmäßig? **Mehrfachnennungen möglich**

- Mit Leuten aus Deutschland ..... 1
- Mit Leuten aus Österreich, Schweiz ..... 2
- Mit Leuten aus Frankreich, Italien, Spanien ..... 3
- Mit Leuten aus Großbritannien, Irland ..... 4
- Mit Leuten aus Osteuropa ..... 5
- Mit Leuten aus Skandinavien ..... 6
- Mit Leuten aus restlichem Europa ..... 7
- Mit Leuten aus USA, Kanada ..... 8
- Mit Leuten aus dem restlichem Amerika ..... 9
- Mit Leuten aus Australien, Neuseeland ..... 10
- Mit Leuten aus der restlichen Welt (z.B.: Asien, Afrika, etc.) ..... 11
- Weiß nicht ..... 12

11. Über welche Medien kommunizieren Sie regelmäßig mit anderen KDElern?

**Mehrfachnennungen möglich**

- Email => weiter mit Frage 11a) ..... 1
- Chat (IRC) => weiter mit Frage 11a) ..... 2
- Telefon => weiter mit Frage 12 ..... 3
- Telefax => weiter mit Frage 12 ..... 4
- Videokonferenzen => weiter mit Frage 12 ..... 5
- Persönliches Treffen => weiter mit Frage 11b) ..... 6
- Weiß nicht => weiter mit Frage 12 ..... 0

11a) Wie nutzen Sie die folgenden Medien durchschnittlich pro Woche in KDE?

**Nur eine Angabe pro Zeile möglich**

	häufig 1	selten 2	nie 3
KDE-Mailinglisten	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
direkte Emails an KDEler	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
KDE-Chat (IRC)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

11b) Zu welchen Gelegenheiten treffen Sie andere KDEler persönlich? **Nur eine Angabe pro Zeile möglich**

	häufig 1	selten 2	nie 3
Bei offiziellen Kongressen (z.B.: Linuxtag)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Bei offiziellen KDE-Treffen (z.B.: vor großen Releases)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Bei inoffiziellen Treffen, um Arbeiten für KDE zu erledigen	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Bei inoffiziellen Treffen aufgrund räumlicher Nähe	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Bei geplanten Besuchs- bzw. Urlaubsreisen, um andere KDEler an ihrem Wohnort zu treffen	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Zufällig	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
sonstige Gründe	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
und zwar: _____			

12. Warum beteiligen Sie sich an KDE? **Nur eine Angabe pro Zeile möglich**

	wichtig 1	Eher wichtig 2	Eher unwichtig 3	unwichtig 4
Störendes Problem für private Anwendung lösen	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Lizenzierung von KDE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Selbstbestätigung	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Der Community etwas zurückgeben	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Erhöhte Aufmerksamkeit in der KDE-Community	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Ablehnung von Microsoft	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Qualifizierung/Weiterqualifizierung	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Intellektuelles Interesse (Problem lösen)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
gemeinsame Arbeit (im Team arbeiten)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Gefühl der Zugehörigkeit zur Community	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Störendes Problem für die Arbeit im Beruf lösen	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Freiheit der Arbeitseinteilung	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Spaß am Programmieren	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Idealismus (politisch, gesellschaftlich, ...)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Sachspenden	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Geld/Entlohnung	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Erwartung der Community	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Freude am Ergebnis	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Erhöhte Aufmerksamkeit der Open source-community (außerhalb von KDE)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Erhöhter Status im Beruf	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Abneigung gegen kommerzielle Software	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Verbesserung der Arbeitsmarktchancen	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Langeweile	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
viel Freizeit	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Hoffnung auf spätere Entlohnung	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Weiterer Grund:	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Und zwar: _____				

13. Haben Sie schon mal für eine Arbeitsleistung bei KDE Geld bekommen?

Ja ...1  Nein ...2  Weiß nicht ...9

Wenn ja, wofür? \_\_\_\_\_

14. Haben Sie schon mal für die Arbeit an KDE eine Sachspende bekommen? Ja ...1  Nein ...2  Weiß nicht ...9

Wenn ja, wofür? \_\_\_\_\_

15. An wievielen KDE-Subprojekten arbeiten Sie derzeit mit?

16. Bei den KDE-Subprojekten: Wieviel andere Leute arbeiten durchschnittlich, maximal und minimal mit?
- |                   |  |
|-------------------|--|
| Durchschnittlich: |  |
| Maximal:          |  |
| Minimal:          |  |

17. Halten Sie Ihre Mitsprachemöglichkeiten an den KDE-Programmiersubprojekten, an denen Sie mitarbeiten, für ... Nur **eine** Angabe möglich
- Hoch ... 1  Eher hoch ... 2  Mittel ... 3  Eher niedrig ... 4  Niedrig ... 5

18. Waren Sie schon an einer Meinungsverschiedenheit in KDE verwickelt?
- Ja ...1  Nein ...2  => weiter mit Frage 19 Weiß nicht ...9

- 18a) Wie oft traten Meinungsverschiedenheiten oder Konflikte in KDE im letzten halben Jahr auf?

- 18b) Wie häufig treten folgende Themen als Konflikte bei KDE auf? Nur **eine** Angabe pro Zeile möglich

	Gar nicht 1	selten 2	häufig 3	Weiß nicht 9
Qualität von programmiertem Code/Übersetzung/Dokumentation	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Art der Kommunikation	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Aufgabenverteilung	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
(verkannte) Anerkennung für geschriebenen Quellcode	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Zukünftige Entwicklungsrichtung von KDE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Zukünftige Ziele eines Subprojekts	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Nichtbeachtung von Beiträgen	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Ablehnung von eingereichtem Quellcode (z.B.: patches)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Löschen oder Verändern von Quellcode	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Selbstüberschätzung der eigenen Leistung	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Anspruchshaltung	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
persönliche Rivalität	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Weiteres Thema:	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Und zwar: _____				

19. Über welche Kontrollart erfolgt im Allgemeinen die Qualitätskontrolle der Quellcodes?

Mehrfachnennungen möglich

- Durch Bugreports ..... 1
- Durch patches ..... 2
- Durch regelmäßige Mitarbeiter am gleichen (Sub-)Projekt ..... 3
- Durch andere KDEler (außerhalb des Subprojekts) ..... 4
- Durch Information von Nutzern/Anwendern in Mailinglisten, direktem Mailkontakt, etc., (nicht über Bugreportssystem) ..... 5
- Durch eigene Feststellung ..... 6
- Durch gegenseitige abgesprochene Überprüfung von Softwareteilen, patches, etc. .... 7
- durch sonstige ..... 8

Und zwar: \_\_\_\_\_



## **Anhang D: Kategorien der Auswertung der Selbstaussagen des KDE-Projekts**

Auf der Homepage wurden von Mitte 2001 bis Mitte 2002 jede Woche ein Email-Interview mit einem Open Source-Beteiligten des untersuchten Open Source-Projekts veröffentlicht. Insgesamt konnten 62 Interviews ausgewertet werden. Diese Interviews sind dabei als Selbstaussagen zu betrachten. Ein KDE-Mitglied erstellte die leicht variierenden Interviewfragebögen. Es kann von vier verschiedenen ähnlichen Fragebögen ausgegangen werden, die sich im Kern um die gleichen Fragen drehten, aber in manchen Themen leicht variierten.

Aus diesen Fragen und den Antworten wurden folgende, z.T. eigene Auswertungskategorien entwickelt:

Alter, Nationalität, Wohnort, Profession, Tätigkeit im KDE-Projekt, Akademiker (abgeschlossenes Studium), Art des Studiums, Studienort, Anstellung im Unternehmen (Anstellung für das Open Source-Projekt), Vollzeitwerbsarbeit/Freizeit, Art der Entlohnung (Firma, Begründung, etc.), Einstiegszeitpunkt, Grund für KDE-Einstieg, Arbeitszeit an dem Open Source-Projekt, Familienstatus, Freund/in, Kinder, Rolle in KDE, eigene derzeitige Projekte/Aufgabe, bisherige Leistungen im KDE-Projekt (Claim to Fame), Probleme/Scheitern/Austritt aus Projekt, Größe des Subprojekts, Face-to-Face-Treffen mit KDElern, Ort des Treffens, Online-/Offline-Freunde, Hobbies/Zeitvertreib, Programmierstil, Sonstiges. Die Kategorien wurden qualitativ-inhaltsanalytisch ausgewertet. Zusätzlich konnten manche Kategorien mittels quantitativer Verfahren ausgewertet werden.