

Otto-von-Guericke-Universität Magdeburg



Thema:

**Entscheidungsmodell für die Migration / Substitution
konventioneller kommerzieller Anwendungssysteme durch Open
Source Software**

Diplomarbeit

Fakultät für Informatik
Arbeitsgruppe Wirtschaftsinformatik

Themensteller: Prof. Dr. Claus Rautenstrauch, Wirtschaftsinformatik (FIN/ITI)

Betreuer: Dipl. Wirtsch.-Inform. Sebastian Herden
Dipl. Wirtsch.-Inform. André Zwanziger

vorgelegt von: Frederik Kramer, Matr.-Nr.: 163784
Winckelmannstraße 12
39108 Magdeburg
frederik.kramer@student.uni-magdeburg.de
kramer@compserve24.de

Abgabetermin: 01. September 2007

Danksagung

Es ist mir ein persönliches Anliegen den Menschen zu danken, welche einen besonderen Anteil an der Vorbereitung und dem Gelingen dieser Arbeit hatten. Das sind zum einen meine Mutter Jutta Kramer, ohne deren antreibende Unterstützung und Motivation ich vermutlich eher ein Opfer der New-Economy als ordentlicher Student der Otto-von-Guericke Universität geworden wäre, sowie mein Vater Dr. Jürgen Kramer, der diese Arbeit dankenswerter Weise mehrfach korrekturgelesen hat und mir damit ein hervorragender Diskussionspartner über die Inhalte der Arbeit wurde.

Meinen Betreuern Herrn André Zwanziger und Herrn Sebastian Herden danke ich für die Betreuung dieser Arbeit. Besonders die fachkundigen, kritischen Augen von Herrn Zwanziger waren mir eine besondere Hilfe und Ansporn während der Erstellung dieser Arbeit. Ferner möchte ich meiner Freundin Thao Ta Minh und Matthias Stern für ihr uneigennütziges, fachfremdes Korrekturlesen, insbesondere im Hinblick auf die neue deutsche Rechtschreibung, danken.

Lizenz

©2007 Frederik Kramer



Das vorliegende Werk ist unter der Creative Commons Attribution-Non-commercial-No Derivative Works 2.0 Germany License lizenziert. Eine Copy der Lizenz kann unter <http://creativecommons.org/licenses/by-nc-nd/2.0/de/> angesehen oder unter Creative Commons, 171 Second Street, Suite 300, San Fransisco, California, 94105, USA per Brief bestellt werden.

Anmerkung

Das Praxisbeispiel (siehe Kapitel 4) dieser Arbeit entstammt einer realen Unternehmenssituation. Zum Schutz des Unternehmens wurden jedoch sämtliche Umsatzzahlen, Namen sowie die Branche verfremdet. Die Zieldefinitonen und die Qualität der Ergebnisse dieses Praxisbeispiels leiden darunter jedoch nicht.

Inhaltsverzeichnis

| | |
|---|-------------|
| Inhaltsverzeichnis | vii |
| Abbildungsverzeichnis | ix |
| Tabellenverzeichnis | xii |
| Abkürzungsverzeichnis | xiii |
| 1 Einführung | 1 |
| 1.1 Einleitung | 1 |
| 1.1.1 Motivation und Ziele | 1 |
| 1.1.2 Freie / Open Source Software | 3 |
| 1.2 Die Geschichte freier Software | 3 |
| 1.2.1 IBM und SHARE | 3 |
| 1.2.2 Unix und die Entstehung der Softwareindustrie | 4 |
| 1.2.3 GNU und Linux | 5 |
| 1.3 Die Ökonomie freier Software | 6 |
| 1.3.1 Wettbewerb und Staat im Softwaremarkt | 7 |
| 1.3.2 Zukunft des Softwaremarktes | 8 |
| 1.4 Open Source Software in der Unternehmenspraxis | 10 |

| | | |
|----------|---|-----------|
| 2 | Theoretische Grundlagen | 13 |
| 2.1 | Grundlegende Begriffe | 14 |
| 2.1.1 | Definitionen | 14 |
| 2.1.2 | Projektmanagement | 17 |
| 2.1.3 | Konsequenzen grundlegender Begriffe | 18 |
| 2.2 | Strategische Faktoren des Einsatzes von OSS | 19 |
| 2.2.1 | Strategische Betrachtung von OSS | 20 |
| 2.2.2 | Open Source Geschäftsmodelle | 21 |
| 2.2.3 | Rechtsgrundlagen des Einsatzes von OSS | 24 |
| 2.2.4 | Konsequenzen strategischer Faktoren | 27 |
| 2.3 | Technische Grundlagen von Anwendungssystemen | 30 |
| 2.3.1 | Substitution / Migration | 30 |
| 2.3.2 | Hardware | 31 |
| 2.3.3 | Software | 33 |
| 2.3.4 | Softwarequalität und Reife von OSS | 35 |
| 2.3.5 | Konsequenzen technischer Grundlagen | 38 |
| 2.4 | Ökonomische Bewertung von AS-Alternativen | 40 |
| 2.4.1 | Kostenbewertung des Einsatzes von OSS | 40 |
| 2.4.2 | Nutzenbewertung des Einsatzes von OSS | 44 |
| 2.4.3 | Wirtschaftlichkeitsbewertung | 48 |
| 2.4.4 | Konsequenzen ökonomischer Bewertungsverfahren | 54 |

| | | |
|----------|---|-----------|
| 3 | Entscheidungsmodell | 57 |
| 3.1 | Projektvorstudie | 58 |
| 3.1.1 | Ermittlung von Projektressourcen | 59 |
| 3.1.2 | Abgrenzung des Untersuchungsobjekts | 61 |
| 3.1.3 | Risikobestimmung des Projektes | 63 |
| 3.1.4 | Bestimmung qualitativer und strategischer Ziele | 65 |
| 3.1.5 | Projektvorstudienbericht | 67 |
| 3.2 | Bestimmung der IST-Situation | 68 |
| 3.2.1 | Hardware | 71 |
| 3.2.2 | Software | 71 |
| 3.2.3 | Prozesse | 72 |
| 3.3 | Alternativenauswahl/-analyse und Machbarkeit | 74 |
| 3.3.1 | Auswahlmaßstab erstellen | 75 |
| 3.3.2 | Alternativen suchen | 75 |
| 3.3.3 | Alternativen bewerten | 77 |
| 3.3.4 | Technische Analyse und Machbarkeit | 78 |
| 3.4 | Qualitativ-strategische Bewertung | 79 |
| 3.5 | Monetäre Bewertung | 81 |
| 3.6 | Entscheidung | 82 |
| 4 | Anwendungsbeispiel des Entscheidungsmodells | 84 |

| | | |
|----------|--|------------|
| 4.1 | Kurzvorstellung der Placeholder GmbH | 84 |
| 4.2 | Projektvorstudienbericht | 85 |
| 4.3 | Bestimmung der IST-Situation | 89 |
| 4.4 | Alternativenauswahl/-analyse und Machbarkeit | 90 |
| 4.4.1 | Erstellen des Auswahlmaßstabes | 91 |
| 4.4.2 | Alternativensuche | 91 |
| 4.4.3 | Alternativenbewertung | 93 |
| 4.4.4 | Technische Analyse und Machbarkeit | 93 |
| 4.5 | Qualitativ-strategische Bewertung | 98 |
| 4.6 | Monetäre Bewertung | 100 |
| 4.7 | Entscheidung | 104 |
| 4.8 | Ergebniszusammenfassung und Diskussion des Modells | 106 |
| 5 | Zusammenfassung und Ausblick | 110 |
| A | Anhang zu Theoretische Grundlagen | 115 |
| A.1 | Antitrust Gesetz in den USA | 115 |
| A.2 | Magisches Dreieck des Projektmanagement | 115 |
| B | Modellhilfsmittel | 117 |
| B.1 | Projektrisiko-Checkliste | 117 |
| B.2 | Bewertung des Risikos und Empfehlung | 119 |
| B.3 | Erhebungshilfen | 121 |

| | | |
|----------|---|------------|
| B.4 | Subversion Aktivitätsvergleich | 138 |
| B.5 | Modellierungen / Placeholder GmbH | 140 |
| C | Umfrage | 143 |
| C.1 | Einleitung Umfrage | 143 |
| C.2 | Ziel der Umfrage | 143 |
| C.3 | Vorgehen bei der Umfrage | 144 |
| C.4 | Ergebnisse der Umfrage | 145 |
| C.4.1 | Allgemeine Ergebnisse | 145 |
| C.4.2 | Unternehmen die OSS einsetzen | 145 |
| C.4.3 | Unternehmen die aktuell keine OSS einsetzen | 149 |
| C.5 | Zusammenfassung der Ergebnisse | 150 |
| C.6 | Kritische Würdigung der Ergebnisse | 150 |
| | Literaturverzeichnis | 151 |

Abbildungsverzeichnis

| | | |
|-----|---|-----|
| 1.1 | Relativer Marktanteil verschiedener Webserver | 11 |
| 2.1 | Abgrenzung Geschäftsprozess / Workflow | 15 |
| 2.2 | Klassifikation von Software | 33 |
| 2.3 | Vorteile von Open Source Software | 37 |
| 2.4 | Herausforderungen in der IT | 41 |
| 2.5 | Total Cost of Ownership | 42 |
| 2.6 | Portfolio | 51 |
| 2.7 | Total Economic Impact | 53 |
| 3.1 | Entscheidungsmodell | 57 |
| 3.2 | Risikobereiche | 63 |
| 3.3 | IT-Strategie - Aufbau | 66 |
| 3.4 | IST-Erhebungsmodell | 70 |
| 4.1 | Beispielportfolio des Migrationsprojekts | 105 |
| 5.1 | Entscheidungsmodell | 111 |
| A.1 | Magisches Dreieck | 116 |

| | | |
|-----|---|-----|
| B.1 | Bewertung des Risikos und Empfehlung | 120 |
| B.2 | Subversion Aktivitätsvergleich | 139 |
| B.3 | Organigramm der Placeholder GmbH | 140 |
| B.4 | Wertschöpfungskette der Placeholder GmbH | 141 |
| B.5 | Prozessmodell der Kundenbestellung bei der Placeholder GmbH | 142 |
| C.1 | Hauptgründe des Interesses an OSS | 146 |
| C.2 | Ausschlaggebende Entscheidungsfragen | 146 |

Tabellenverzeichnis

| | | |
|------|---|-----|
| 2.1 | Verteilung der IT-Projektergebnisse in amerikanischen Unternehmen | 17 |
| 2.2 | Punkteskala | 46 |
| 4.1 | Bewertung der ERP-Alternativen | 93 |
| 4.2 | Funktionale Kriterien für AS-Alternativen | 94 |
| 4.3 | Übersicht technischer Details | 96 |
| 4.4 | Bewertungsschlüssel für die Kriterienerfüllung | 98 |
| 4.5 | Erfüllung der Kriterienliste durch die Alternativen | 98 |
| 4.6 | Zielgewichtung qualitativ-strategischer Ziele | 99 |
| 4.7 | Qualitativ-strategische Alternativenbewertung | 100 |
| 4.8 | Kosten der Einführung von AS-Alternativen | 102 |
| 4.9 | Kosten des Betriebes von AS-Alternativen | 103 |
| 4.10 | Nutzen aus dem Betrieb von AS-Alternativen | 103 |
| 4.11 | Risikowert der AS-Alternativen | 104 |
| B.1 | Projektrisiko-Checkliste | 118 |
| B.2 | Server Erhebungshilfe | 122 |
| B.3 | Clients Erhebungshilfe | 123 |

| | | |
|------|--|-----|
| B.4 | Netzwerk Erhebungshilfe | 124 |
| B.5 | Druck Erhebungshilfe | 125 |
| B.6 | Sonstige Erhebungshilfe | 126 |
| B.7 | Betriebssysteme Erhebungshilfe | 127 |
| B.8 | Serverdienste Erhebungshilfe | 128 |
| B.9 | Standardsoftware Erhebungshilfe | 129 |
| B.10 | Individualsoftware Erhebungshilfe | 130 |
| B.11 | Dokumentenvorlagen und Makros | 131 |
| B.12 | Workflows - Architektur und Nutzer | 132 |
| B.13 | Workflows - Client-/ Serverbetriebssysteme | 133 |
| B.14 | Workflows - Datenbanksysteme, Applikationsserver | 134 |
| B.15 | Workflows - Benutzer-/ Rechteverwaltung und Schnittstellen . | 135 |
| B.16 | Workflows - Verantwortung | 136 |
| B.17 | Workflows - Ausblick | 137 |
| | | |
| C.1 | Umfrage - Branchenzusammensetzung | 145 |
| C.2 | Umfrage - Hauptgründe des Interesses an OSS | 146 |
| C.3 | Umfrage - Entscheidungsfragen | 147 |
| C.4 | Umfrage - Einsatzart | 148 |
| C.5 | Umfrage - Softwareklasse | 148 |
| C.6 | Umfrage - Software | 149 |

| | |
|----------------------------------|-----|
| C.7 Umfrage - Analysen | 149 |
|----------------------------------|-----|

Abkürzungsverzeichnis

| | |
|-------------|---|
| AHP | Analytical Hierarchy Process |
| ARPA | Advanced Research Project Agency |
| AS | Anwendungssystem |
| ASP | Application Service Providing |
| AT&T | American Telephone & Telegraph Corporation |
| BMI | Bundesministerium des Inneren |
| BSD | Berkeley Software Distribution License |
| BStU | Die Bundesbeauftragte für die Unterlagen des Staatssicherheitsdienstes der ehemaligen Deutschen Demokratischen Republik |
| CMM | Capability Maturity Modell |
| CSS | Closed Source Software |
| CTI | Computer Telephony Integration |
| DRM | Digital Rights Management |
| EAI | Enterprise Application Integration |
| ERP | Enterprise Resource Planning |
| FSF | Free Software Foundation |
| FUD | Fear, Uncertainty, Doubt |
| GDPdU | Grundsätze zum Datenzugriff und zur Prüfbarkeit digitaler Unterlagen |
| GNU | GNU's not Unix |
| GoB | Grundsätze ordnungsgemäßer Buchführung |
| GoBS | Grundsätze ordnungsgemäßer DV-gestützter Buchführungssysteme |
| GPL | General Public License |
| HIPAA | Health Insurance Portability and Accountability Act |
| HPC | High Performance Computing |
| HTTP | Hypertext Transfer Protocoll |
| IBM | International Business Machines Cooperation |
| IDA | Interchange of Data and Administrations |
| IDABC | Interoperable Delivery of European eGovernment Services to public Administrations, Businesses and Citizens |
| ISB | Schweizerisches Informatikstrategieorgan Bund |
| JRE | Java Runtime Environment |

| | |
|---------------|---|
| KBSt | Koordinations- und Beratungsstelle für Informations- technik in der Bundesverwaltung |
| KMU | kleine und mittelständische Unternehmen |
| KonTraG | Gesetz zur Kontrolle und Transparenz im Unterneh- mensbereich |
| LAN | Local Area Network |
| LGPL | Lesser General Public License |
| MIT | Massachusetts Institute of Technology |
| MVC | Model View Controller |
| OSI | Open Source Initiative |
| OSS | Open Source Software |
| OTRS | Open Ticket Request System |
| PDF | Portable Document Format |
| RDBMS | Relationales Datenbank Management System |
| SMTP | Send Mail to Port |
| SOA | Service Oriented Architecture |
| SOAP | Simple Object Access Protocoll |
| SOX | Sarbanes-Oxley Act |
| SPICE | Software Process Improvement and Capability Determi- nation |
| SVN | Subversion |
| TEI | Total Economic Impact |
| TMG | Telemediengesetz |
| TQM | Total Quality Management |
| UUCP | Unix to Unix Copy |
| WMS | Workflow Management System |
| XML | Extensible Markup Language |

Kapitel 1

Einführung

1.1 Einleitung

Die Fähigkeit zu flexiblem Wandel und kurze Produktlebenszyklen kennzeichnen die IT-Industrie. Informationstechnologie hat in der Vergangenheit erhebliche Auswirkungen auf die Wirtschaft gehabt und sie dabei nachhaltig verändert.

Laut Porter (vgl. Porter, 1998, S. 76) beeinflusst Informationstechnologie den gesamten Wertschöpfungsprozess eines Unternehmens und damit sein Waren- und Dienstleistungsportfolio. Kein Wirtschaftssektor kommt heute ohne leistungsfähige Informationssysteme aus. Fast immer gehören sie zu den entscheidenden Schlüsselfaktoren bei der Effizienzsteigerung von Geschäftsprozessen und tragen somit zum Erfolg eines Unternehmens bei.

Die „*Open Source Software Bewegung*“ ist in den vergangenen zehn Jahren zu einem Technologietreiber der IT-Industrie geworden.

1.1.1 Motivation und Ziele

Die im Rahmen meiner praktischen Arbeit erworbenen Erfahrungen beim Einsatz von Open Source Software (OSS) haben die Frage nach einem theo-

retisch fundierten und gleichzeitig praktisch anwendbaren Entscheidungsmodell zu deren Einsatz im Unternehmensumfeld aufgeworfen.

Die mit dieser Arbeit durchgeführte Studie (siehe Anhang C) konnte im Hinblick auf die Verbreitung von OSS in Unternehmen (85% der befragten Unternehmen setzen demnach bereits OSS ein) darüber hinaus die positiven Ergebnisse andere Studien bestätigen (vgl. Lykkegaard, 2005; Wichmann, 2002; Méndez et al., 2005; Walli et al., 2005).

Der produktive Einsatz von Open Source Software beruht häufig auf wenig geprüften strategischen Überlegungen, einseitigen Kostenbetrachtungen (vgl. Ghosh, 2006, S. 238 ff.) oder schlicht unternehmerischen Bauchentscheidungen. Häufig wird Open Source Software im kleinen Maßstab erprobt und eingesetzt (siehe Anhang C). OSS wird teilweise sogar ohne das Wissen von Entscheidern zur Lösung individueller Probleme eingesetzt (vgl. Wichmann, 2004, S. 33).

Dies führt mit steigender Größe des Unternehmens zu einem unüberschaubaren Wildwuchs an Anwendungssystemen und damit zu besonders teuren und risikobehafteten Migrationsprojekten. Unsystematisches Vorgehen gefährdet den erfolgreichen Einsatz von Open Source Software und damit den langfristigen Erfolg einer darauf ausgerichteten IT-Strategie. Nur mit sorgfältiger technischer und ökonomischer Prüfung lässt sich der langfristig erfolgreiche Einsatz freier Software erreichen.

Ziel der Arbeit ist ein Entscheidungsmodell zu entwickeln, das alle wesentlichen Elemente einer ökonomisch und technisch fundierten Entscheidung beinhaltet und möglichst einfach anwendbar ist. Das heißt, der Aufwand der Entscheidungsvorbereitung muss relativ zum erwarteten Projektvolumen gering sein. Insbesondere soll eine reine Kostenperspektive, wie sie die oft angewandte TCO-Analyse (vgl. Brügge et al., 2004, S. 116 ff.) bietet, vermieden werden. Nach einer allgemeinen Einführung in das Thema Open Source Software und deren Geschichte, werden zunächst die theoretischen Grundlagen des Entscheidungsmodells erläutert. Das danach entwickelte Entscheidungs-

modell soll beratenden Dienstleistern und Entscheidern eine faktenbasierte Entscheidung über den Einsatz von OSS ermöglichen.

Die beispielhafte Anwendung des erarbeiteten Entscheidungsmodells soll die Anwendbarkeit des Modells im konkreten Praxisfall zeigen und im Rahmen einer kritischen Würdigung Anhaltspunkte zur weiteren wissenschaftlichen Arbeit liefern.

1.1.2 Freie / Open Source Software

In Verbindung mit Open Source Software wird häufig synonym von „freier Software“ gesprochen. Die unterschiedlichen Begriffe spiegeln einen unterschiedlichen Blickwinkel der „*Free Software Foundation*“ (FSF) auf der einen und der „*Open Source Initiative*“ (OSI) auf der anderen Seite auf das Phänomen OSS wider und resultieren aus der Fehlinterpretierbarkeit des Wortes „free“. Im englischen Sprachgebrauch bedeutet „free“ sowohl frei als auch kostenlos.

„Freie Software“ und „Open Source Software“ werden im Rahmen dieser Arbeit synonym verwandt.

1.2 Die Geschichte freier Software

1.2.1 IBM und SHARE

Der Zeitraum zwischen Anfang der 1960er Jahre und Anfang der 1980er Jahre kann als Geburtsstunde „*Freier Software*“ (Der Begriff „Open Source Software“ entstand erst 1998 (vgl. Andreessen, 1998; Wichmann, 2004, , S. 10)) betrachtet werden. Die Firma IBM, die 1959 95% des IT-Marktes (vgl. Mühlbauer, 2000) kontrollierte, erwirtschaftete mit der Leihstellung ih-

rer Großrechner Gewinne und lieferte notwendige Software ohne gesonderte Berechnung. Erst ein Kartellrechtsstreit zwang IBM erstmalig Computersysteme zu verkaufen, die bis 1952 lediglich vermietet worden waren (vgl. Mühlbauer, 2000).

Bereits Ende der 1950er Jahre hatte sich der IBM-Anwenderclub SHARE formiert, dessen Zweck die Effizienzsteigerung der Ressourcen beim Arbeiten mit IBMs Computersystemen war. Darüber hinaus sollte erreicht werden, dass IBM mehr auf Nutzerbedürfnisse eingeht (vgl. Leonard, 2000). Durch die bei SHARE übliche Praxis, Code für das System 701 von IBM zu tauschen, entstand damals ein Vorläufer der heutigen „*Open Source Software Bewegung*“ unter den Nutzern dieser Maschinen.

Die Softwareindustrie entwickelte sich jedoch erst, als im Zuge eines Anti-Trust Verfahrens gegen IBM (siehe Abschnitt A.1) Hardware, Software und Dienstleistung entbündelt werden mussten.

1.2.2 Unix und die Entstehung der Softwareindustrie

Auf Grund mangelnder Rechenleistung galt es bis Ende der 1970er Jahre als nahezu unmöglich, komplette Betriebssysteme in Hochsprachen zu entwickeln und damit den ersten Schritt zur Plattformunabhängigkeit zu vollziehen. Hardware ließ sich nur mit eigens entwickelten Betriebssystemen verkaufen. Seit Anfang der 1960er Jahre arbeiteten verschiedene Firmen an der Entwicklung eines plattformübergreifenden Betriebssystems. AT&T veröffentlichte schließlich 1969 das erste proprietäre assemblerbasierte UNIX. Universitäten und andere Forschungseinrichtungen, die das Betriebssystem nahezu lizenzkostenfrei erwerben konnten, begannen Basisdienste für ein weltumspannendes Computernetzwerk zu entwickeln.

Das von der Advanced Research Project Agency (ARPA) im Rahmen dezentraler Verteidigungsstrategien ins Leben gerufene ARPANet sorgte zeitgleich

dafür, dass der Tausch von Sourcecode in Grenzzeit unter angeschlossenen Systemen möglich und üblich wurde. Nachdem 1978 die vollständige Portierung von UNIX auf die Programmiersprache C realisiert war (vgl. Raymond, 2001, S. 8), sorgte das Unixprotokoll UUCP für die schnelle Ausbreitung des darauf basierenden Usenet¹, was den Austausch von Informationen fortan erheblich begünstigte (vgl. Raymond, 2001; Wichmann, 2004, S. 4 f., S. 8-11).

1.2.3 GNU und Linux

Ende der 1980er Jahre änderte AT&T seine Lizenzpolitik und andere Hardwarehersteller begannen daraufhin damit, eigene Unixderivate zu entwickeln.

Die anhaltende Kommerzialisierung von UNIX bewegte Richard Stallmann, der damals am Massachusetts Institute of Technology (MIT) angestellt war, 1984 die Free Software Foundation zu gründen und seine Arbeiten an seinen Unix Kommandozeilentools (GNU Tools) unter der von ihm entwickelten General Public License (GPL) zu veröffentlichen.

Linus Torvalds, damals wissenschaftlicher Mitarbeiter an der Universität von Helsinki, veröffentlichte 1991 seine Arbeiten an einem unixähnlichen Betriebssystemkern².

Das Entwicklungsparadigma „*Release early and often*“ (vgl. Raymond, 2001, S. 28 ff.) und die Art, wie Torvalds andere Entwickler in das Projekt mit einbezog, sorgten bald nach Veröffentlichung für eine große Zahl an Verbesserungen und Feedback interessierter Softwareentwickler. Raymond (vgl. Raymond, 2001, S. 30) umschrieb diesen strikt auf „Entwicklung“ und „Fehlersuche“ ausgerichteten Prozess mit: „*Given enough eyeballs, all bugs are shallow*“.

¹ oft als das erste digitale Schwarze Brett bezeichnet

² <http://groups.google.de/group/comp.os.minix/msg/b813d52cbc5a044b?dmode=source>, besucht am 26.08.2007

Linux erreichte, begünstigt durch die zunehmende Verbreitung des Internet (vgl. Lerner und Tirole, 2005, S. 52), bereits Ende 1993 eine Reife, die es mit kommerziellen Unixderivaten standhalten ließ. Dennoch dauerte es fünf Jahre, bis eine breite Öffentlichkeit erreicht wurde und die neue Technologieära weithin wahrgenommen wurde (vgl. Raymond, 2001, S. 16).

1.3 Die Ökonomie freier Software

Das Phänomen „*Open Source Software*“ ist mit der klassischen Betriebswirtschaftslehre nur schwer erklärbar. Laut Perens (vgl. Perens, 2005) ist nicht sofort klar, wie OSS ökonomisch zu erklären ist. Dies sorgt für Unsicherheit bei verantwortlichen IT-Entscheidern. In der Theorie des „homo oeconomicus“ wird davon ausgegangen, der Mensch verhalte sich stets ökonomisch rational. Eine unentgeltliche Beteiligung an der Entwicklung eines OSS-Projektes scheint damit zunächst im Widerspruch zu stehen, da keine direkte monetäre Kompensation für das erforderte Arbeitsleid geleistet wird.

Nach der von Ghosh (vgl. Ghosh, 2005, S. 23 ff.) vorgestellten Umfrage zu den Motiven von Open Source Entwicklern, stehen Lernen, Wissensteilung und Partizipation am OSS-Phänomen mit 53% im Vordergrund. Circa 31% der Befragten geben Karriereziele und monetäre Belange (vor allem Signaling Effekte, wie Reputation und verbesserte Jobchancen) als Motivation an. Circa 12% nennen politische Motivation (Software sollte „frei“ sein, Schwächung von großen Softwarekonzernen) als Grund für ihre Teilnahme. Lediglich für 2,6 Prozent der Befragten standen Softwarespezifika (z. B. Verbesserung einer speziellen Software) im Vordergrund.

Dennoch resultiert die Entwicklung von OSS zumindest zu einem Teil aus ökonomischen Motiven. Nach einer von Ghosh (vgl. Ghosh, 2006, S. 44 ff.) veröffentlichten Untersuchung der Codebasis des Debian Projekts beteiligten sich Firmen mit insgesamt 1,2 Milliarden € an der Entwicklung freier

Software und erzielten damit Zusatzerträge in Höhe von 263 Millionen €. Allein Sun Microsystems investierte bis zum Jahr 2006 312 Millionen € in die Entwicklung freier Software (vgl. Ghosh, 2006, S. 51).

1.3.1 Wettbewerb und Staat im Softwaremarkt

Der unternehmerische Anreiz Software zu produzieren ist hoch. Hohe Fixkosten und niedrige variable Kosten sorgen für hohe Skalenerträge. Positive externe Effekte durch große Anwenderbasen sorgen für mögliche „lock-in“-Effekte und daher hohe Marktkonzentration (natürliche Monopole). Hohe Wechselkosten machen den Markt zusätzlich ineffizient (vgl. Brügge et al., 2004, S. 126).

Wiederholt waren auf Basis des Antitrust Gesetzes (siehe Anhang A.1) staatliche Eingriffe in das Marktgefüge der Softwareindustrie notwendig. Das Antitrust Verfahren (siehe Anhang A.1) der US Justizbehörde gegen Microsoft³ wurde erst nach rund sechs Jahren durch Bestätigung einer außergerichtlichen Einigung beigelegt (vgl. Gonzales, 2001; Heise, 2004).

Die Beteiligung der Staaten am Softwaremarkt und die Haltung gegenüber dem Open Source Phänomen ist daher viel diskutiert. Staaten haben in Zeiten knapper öffentlicher Kassen einen hohen Anreiz, vermeintliche Kosteneinsparpotenziale in Verbindung mit dem Einsatz von OSS wahrzunehmen. Staatliches Eingreifen in Märkte, egal ob als Abnehmer oder als Anbieter, ist immer vor dem Hintergrund staatlicher Wohlfahrtsoptimierung zu bewerten.

Comino und Manenti (Comino und Manenti, 2004) zeigen, dass staatliche Informationskampagnen zur Verhinderung asymmetrischer Informationsverteilung sowie der rechtsverbindliche Einsatz von OSS unter der Annahme, dass die Maßnahmen eine große Zahl Anwender zu deren Einsatz bewegen

³ wegen des, aus Sicht der US Behörde wettbewerbswidrigen „Bundeling“ des Internet-Explorer mit dem Betriebssystem Windows

können, gesamtwirtschaftlich wohlfahrtsoptimierend sind. Allein die Subventionierung würde zu einem Wohlfahrtverlust führen.

Kooths et al. (vgl. Kooths et al., 2003, S. 6 f.) halten die Beteiligung des Staates am Softwaremarkt lediglich als Nachfrager und auch dann nur auf Grundlage wirtschaftlicher Analysen für geeignet, da der Staat nicht über die Voraussetzungen verfügt, die staatliches Eingreifen rechtfertigen würden (z. B. Kenntnis über zukünftige Marktentwicklung und -struktur).

Tatsächlich forcieren neun Länder der Europäischen Union⁴- unter ihnen auch die Bundesrepublik Deutschland⁵- sowie neun weitere Staaten den Einsatz von Open Source Software (vgl. Lee, 2006, S. 60). Peru hat sogar ein Gesetz zum bevorzugten Einsatz von OSS verabschiedet (vgl. Kellermann, 2005).

1.3.2 Zukunft des Softwaremarktes

Prognosen über die Zukunft des Softwaremarktes lassen sich nur schwer anstellen. Sicher ist, dass Open Source Software externe Effekte auf die Anbieter proprietärer Software ausübt. Wie bereits erwähnt neigt der Softwaremarkt zur Konzentration (siehe Abschnitt 1.3.1) und damit zur Standardisierung (das PDF-Format von Adobe mag als Beispiel dienen). Dabei kommt es wesentlich darauf an, wie groß die Nutzerzahl eines Produktes ist und ob die Kostenvorteile von Open Source Produkten⁶ auch in Zeiten der Kommerzialisierung von OSS (siehe Abschnitt 2.2.2) Bestand haben. Franck und Jungwirth (vgl. Franck und Jungwirth, 2001, S. 14 ff.) kommen bei ihrer Betrachtung der Betriebssysteme Linux und Windows zum Schluss, dass die Hold-up Strategie von Microsoft neben der OSS Entwicklergemeinde auch Konkurrenten wie HP und IBM zur Intensivierung ihrer Investitionen in Linux bewegt. Microsoft zwingt seine große Nutzerbasis immer wieder zu

⁴ <http://ec.europa.eu/idabc/en/document/2627/5894>, besucht am 02.07.2007

⁵ <http://www.bundestux.de>, besucht am 02.07.2007

⁶ in Form unentgeltlicher Entwicklungsleistung

weiteren Investitionen in Hard- und Software und Komplementhersteller zur Adaption proprietärer Schnittstellen und der eigenen Innovationsgeschwindigkeit.

Für den Versuch einer Zukunftsprognose soll an dieser Stelle auf Fitzgerald (vgl. Fitzgerald, 2005, S. 93 ff.) verwiesen werden. Der OSS-Bewegung muss es demnach u. a. gelingen, inhärente Probleme wie das Aussterben der OSS-Pioniere⁷, fehlende Geschäfts- und Marketingstrategien, Aufrechterhaltung der Werte u. a. zu lösen (vgl. Fitzgerald, 2005, S. 93 ff.). Insbesondere die von der FSF geprägte Ideologie freier Software

„Free software“ is a matter of liberty, not price. To understand the concept, you should think of „free“ as in „free speech“, not as in „free beer“ (vgl. Stallman, 2007)

wird in eine für alle Interessengruppen akzeptable Darstellung zu wandeln sein, die weniger ideologisch, dafür pragmatischer die Vorteile von OSS darstellt, damit diese im globalen Softwaremarkt nachhaltig bestehen oder diesen sogar nachhaltig verändern kann.

Neben der pragmatischen Darstellung qualitativer und strategischer Vorteile des Einsatzes wird die rechtliche Situation in Verbindung mit der Nutzung von OSS in Zukunft richtungsbestimmend sein. Vor allem die juristische Fachdiskussion um die grundsätzliche Wirksamkeit der Lizenzen, Fragen der Produkthaftung und Gewährleistung sowie der Einführung und Wirkung von Softwarepatenten muss geführt werden und schließlich zu einem klaren Rechtsrahmen führen (siehe Abschnitt 2.2.3).

Die Anbieter proprietärer Software werden die Vorteile ihrer Produkte deutlicher, auch gegenüber der OSS-Konkurrenz, darstellen müssen. Die Unabhängigkeit der jüngeren Microsoft Kampagne „Get the facts“ wird stark bezweifelt (vgl. Vaughan-Nichols, 2004). Dazu tragen die unvollständige

⁷ z.B. Torvalds, Behlendorf, Tridgell, Stallmann u. a.

Verwendung von TCO-Analysen (siehe Abschnitt 2.4.1) oder weitgehend undurchschaubare Benchmarktests sogenannter FUD-Strategien (FUD steht für Fear, Uncertainty, Doubt) bei.

Open Source Software findet immer häufiger auch das Interesse von Risikokapitalgebern (vgl. Heise, 2007), so dass Wettbewerb mittelfristig nicht mehr durch knappe Werbebudgets oder fehlende Organisation entschieden wird, sondern mehr auf der Ebene überprüfbarer Qualitäts- und Wirtschaftlichkeitsbetrachtungen ausgetragen wird.

1.4 Open Source Software in der Unternehmenspraxis

„*Linux*“ ist die bekannteste Open Source Software⁸ (vgl. Wichmann, 2004, S. 14). Im Serverbereich ist es bereits fester Bestandteil der IT-Infrastruktur vieler Unternehmen und stellt nach Windows die zweithäufigste Betriebssystemfamilie (vgl. Saleck und Safari-Fard, 2005, S. 93) dar. Analysten von IDC ermittelten 2005 bereits einen über 40%igen Einsatz von OSS als Serverbetriebssystem (vgl. Ghosh, 2006, S. 20).

Fälschlicherweise werden Linux und OSS häufig synonym verwendet (vgl. Saleck und Safari-Fard, 2005, S. 18). „Open Source Software“ umfasst jedoch weit mehr als nur Linux. Sourceforge, die weltweit größte Plattform für OSS, listet zur Zeit ca. 150.000 Projekte⁹. Nicht jedes Projekt bringt dabei für den Unternehmenseinsatz geeignete oder brauchbare Software hervor. Otis et al. (Otis et al., 2007) liefern einen sehr detaillierten Katalog mit bereits im Unternehmensumfeld getesteter und genutzter OSS. Manche OSS-Produkte sind darüber hinaus bereits zum Industriestandard in ihrem Anwendungsfeld geworden, wie zum Beispiel Abbildung 1.1 zeigt.

⁸ siehe Anhang C

⁹ Stand 01.07.2007

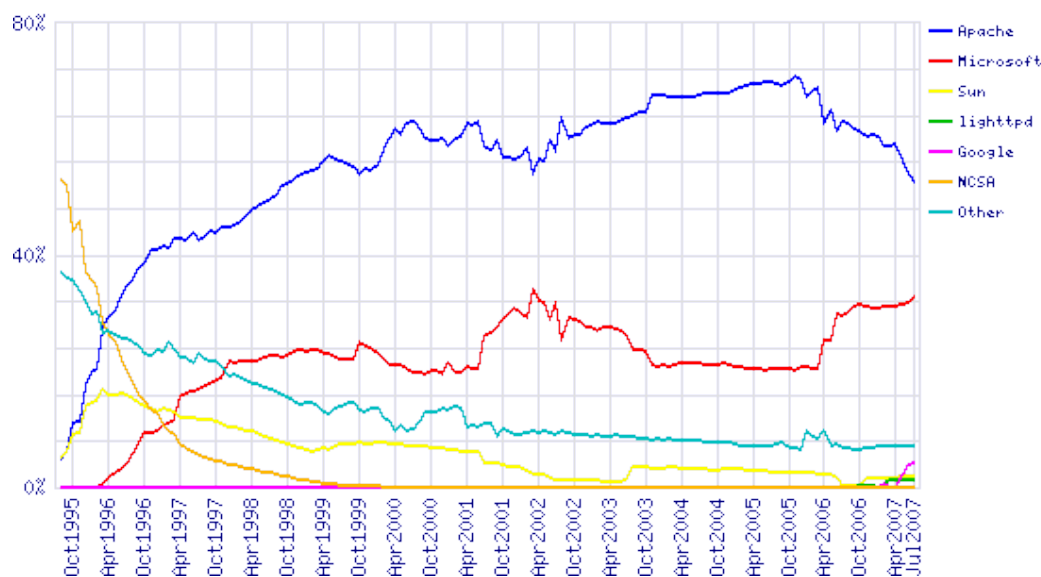


Abbildung 1.1: Relativer Marktanteil verschiedener Webserver (vgl. Netcraft, 2007)

Eine im Jahr 2005 von Forrester Research (vgl. Méndez et al., 2005) durchgeführte Studie ermittelte, dass 40% der europäischen Unternehmen und öffentlichen Einrichtungen bereits 2005 OSS einsetzten und weitere 8% den Ersteinsatz für 2006 geplant hatten. Immerhin 45% der Organisationen setzten 2005 OSS bereits für geschäftskritische Anwendungen ein.

Entscheidern stellt sich dabei grundsätzlich die Frage, wie zu den jeweiligen Bedürfnissen passende OSS ausgewählt wird, und ob die betrachtete OSS für das Unternehmen¹⁰ überhaupt eine gangbare Alternative zu bekannter Closed Source Software (CSS) darstellt. Dabei sollte die Entscheidung über den Einsatz einer Software, egal ob Open Source oder Closed Source, stets auf der Ebene fundierter wirtschaftlicher, technischer sowie strategischer Analysen getroffen werden. Die vorliegende Arbeit wird dazu einen Beitrag leisten.

Eine 2005 von der Fraunhofergesellschaft durchgeführte Selbstuntersuchung ergab, dass OSS vor allem in den Bereichen Betriebssysteme, Office-

¹⁰ oder die Behörde

Produkte, Datenbanken und Entwicklungssysteme im direkten Vergleich zu CSS bereits als „reif“ (siehe Abschnitt 2.3.4) betrachtet wird. Eingesetzt wird OSS bei der Fraunhofergesellschaft im direkten Vergleich mit CSS vorwiegend im Bereich der Datenbanksysteme, Entwicklersysteme, Webserver, Webbrowser, Content Management System und Grid bzw. Clustersoftware (vgl. Renner et al., 2005, S. 68).

Kapitel 2

Theoretische Grundlagen

Auf Grund ihrer weitreichenden Bedeutung für die Wertschöpfung und den Haushalt eines Unternehmens, ist die Entscheidung über den Einsatz von Anwendungssystemen eine echte Führungsaufgabe. Um diese Führungsaufgabe in optimaler Weise ausüben zu können, ist die Kenntnis aller strategisch relevanten Merkmale des Einsatzes der betreffenden Informationstechnologie ebenso wichtig wie die Kenntnis ihrer technischen Grundlagen und der ökonomischen Konsequenzen.

Dieses Grundlagenkapitel gliedert sich deshalb in einen Abschnitt „*Grundlegende Begriffe für den Einsatz von AS*“ und die drei Hauptabschnitte „*Strategische Faktoren des Einsatzes von OSS*“, „*Technische Grundlagen von Anwendungssystemen*“, „*Ökonomische Bewertung von AS-Alternativen*“. Im Rahmen der „*Grundlegende[n] Begriffe des Einsatzes von Anwendungssystemen*“ (siehe Abschnitt 2.1) werden zunächst Begriffe erläutert, deren exaktes Verständnis für das zu erarbeitende Entscheidungsmodell und dessen exemplarische Anwendung notwendig sind. Darauf folgend wird die Notwendigkeit eines begleitenden Projektmanagements dargestellt.

Jede Entscheidung über den Einsatz von Anwendungssystemen ist nicht nur rein technisch oder ökonomisch bewertbar, sondern sollte im Rahmen unternehmerischen Handelns in entsprechende Strategien gebettet sein. Dafür ist die Kenntnis der strategischen Faktoren des Einsatzes von OSS notwendig, die im ersten Hauptabschnitt dargestellt werden (siehe Abschnitt

2.2). Im zweiten Hauptabschnitt werden die technischen Grundlagen von Anwendungssystemen (siehe Abschnitt 2.3) und im dritten Hauptabschnitt die ökonomische Bewertung von AS-Alternativen (siehe Abschnitt 2.4) dargestellt. Am Ende jedes Hauptabschnittes werden die Implikationen für das Entscheidungsmodell zusammenfassend dargestellt.

2.1 Grundlegende Begriffe des Einsatzes von Anwendungssystemen

2.1.1 Definitionen

Anwendungssystem

Laut Stahlknecht und Hasenkamp (vgl. Stahlknecht und Hasenkamp, 2005, S. 204) umfassen Anwendungssysteme (AS) im engeren Sinne die Gesamtheit der Daten und Anwendungsprogramme für ein konkretes betriebliches Anwendungsgebiet und Anwendungssysteme im weiteren Sinne zusätzlich die notwendige Hardware und sonstige Kommunikationseinrichtungen.

Findet der Begriff „*Anwendungssystem*“ in dieser Arbeit Verwendung, ist stets Anwendungssystem im weiteren Sinne gemeint, da gerade bei OSS-Projekten das Zusammenspiel von Hardware und Software beachtet werden muss, weil nicht jede Hardwarekomponente zwangsläufig auch mit freier Software betrieben werden kann. Darüber hinaus findet OSS im Zusammenhang mit der kostengünstigen und funktionalen Integration von verschiedensten Kommunikationskanälen in die Informationsinfrastruktur von Unternehmen Anwendung, wie etwa die OSS-Projekte Asterisk oder das Open Ticket Request System (OTRS) zeigen.

Prozess und Workflow

In der Literatur existiert keine einheitliche Definition des „Geschäftsprozesses“. In dieser Arbeit soll unter dem Begriff Geschäftsprozess daher das in der Wirtschaftsinformatik häufig angewandte Begriffsverständnis des „Geschäftsprozesses“ als inhaltlich abgeschlossene, zeitlich-sachlogische Abfolge von Funktionen verstanden werden, die zur Bearbeitung eines für die Leistungserbringung des Unternehmens relevanten Objekts erforderlich ist (vgl. Wysusek, 2001; Rosemann, 1996, S. 210, S. 27).

Ein Workflow (engl. Arbeitsablauf) beschreibt im Gegensatz zu einem Geschäftsprozess genau die auszuführenden Arbeitsschritte mit dem Ziel des (teil-)automatisierten Ablaufs. Ein Geschäftsprozess beschreibt also „was“ zu tun zu ist, von wem die Inputs kommen und wem die Outputs übermittelt werden, während ein Workflow das „wie“ der operationalen Aufgabenausführung festlegt (vgl. Freund, 2007).

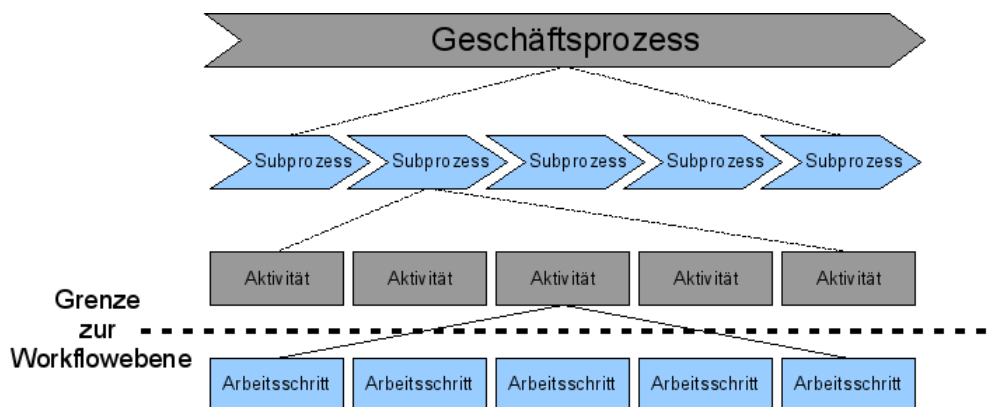


Abbildung 2.1: Eigene Darstellung in Anlehnung an Freund (vgl. Freund, 2007) - Abgrenzung Geschäftsprozess / Workflow

Abbildung 2.1 zeigt, dass Geschäftsprozessmodelle vor allem dazu Verwendung finden können, nach einem „Top-Down“ Vorgehen die betrieblichen Strukturen und deren Interdependenzen aufzuzeigen und diese in Subprozessen zu verfeinern. Die letztliche Tätigkeitsbeschreibung der untersten Prozess-

ebene stellen dabei die Workflows da. Das Verständnis der Geschäftsprozesse und der grundsätzlich automatisierbaren Workflows innerhalb eines Unternehmens sind unbedingte Voraussetzungen für jedes erfolgreiche IT-Projekt.

Qualität

Der Begriff „Qualität“ ist im Kontext technischer Projekte wichtig, weil sich Qualität nach Festlegung geeigneter Metriken im Rahmen des sogenannten produktbezogenen Qualitätsansatzes (vgl. Garvin, 1984) operationalisieren und damit messen lässt. Qualität hat dabei eine entscheidungsrelevante Bedeutung. Diese Bedeutung ist in der neueren Managementliteratur und im besonderen Kontext des „*Total Quality Management (TQM)*“ unumstritten.

Garvin (vgl. Garvin, 1984) differenziert den Qualitätsbegriff in den transzendenten, produktbezogenen, kundenbezogenen, wertorientierten und fertigungsbezogenen Qualitätsbegriff. Bei der qualitativen Betrachtung der Software und der Reife von OSS (siehe Abschnitt 2.3.4) finden der produkt- und fertigungsbezogene Qualitätsbegriff Verwendung. Für die Entscheidungsvorbereitung – und daraus resultierend die Entscheidung selbst – wird in dieser Arbeit jedoch gleichermaßen hohe Qualität gefordert. Für die Wahrnehmung der Führungsaufgabe sind damit der kunden-¹ und wertorientierte² Qualitätsbegriff relevant.

Qualität war in der Vergangenheit laut DIN ISO 8402 als „die Gesamtheit von Merkmalen einer Einheit bezüglich ihrer Eignung, festgelegte und vorausgesetzte Erfordernisse zu erfüllen“, definiert (vgl. Gaulke, 2004, S. 99).

Mit der Einführung der DIN ISO 9000 Normenreihe wurde sie als „der Grad, in dem ein Satz inhärenter Merkmale Anforderungen erfüllt“, definiert und wird in dieser gleichbedeutenden Definition auch in dieser Arbeit verwendet.

¹ damit sind die Stakeholder eines Unternehmens gemeint

² im Sinne eines betriebswirtschaftlichen Optimierungskalküls

2.1.2 Projektmanagement

Bei der Einführung neuer Anwendungssysteme müssen eine Reihe von Entscheidungsdeterminanten geprüft werden, um fundierte und zugleich risikoarme Entscheidungen treffen zu können. Durch das Projektmanagement kann die Gefahr von Dead Ends³ oder kompletten Projektfehlschlägen erheblich verringert werden.

Die Standish Group International Inc. ermittelte in ihrem erstmals 1995 veröffentlichten Chaos-Report⁴ 31% vorzeitig beendete, 52,7% hinsichtlich ihrer Projektziele (siehe Anhang A.2) nicht erfolgreiche und 16,2% erfolgreich durchgeführte Projekte für das Jahr 1994. Die Standish Group schätzte, dass alleine die Kosten der 1995 fehlgeschlagenen Projekte etwa 81 Milliarden Dollar ausgemacht haben (vgl. Gaulke, 2004, S. 39).

| Projektergebnis | 1994 | 1996 | 1998 | 2000 |
|---------------------------------------|------|------|------|------|
| Erfolgreich / Succeed | 16% | 27% | 26% | 28% |
| Ziele teilweise verfehlt / Challenged | 53% | 33% | 46% | 49% |
| Erfolglos / Failed | 31% | 40% | 28% | 23% |

Tabelle 2.1: Verteilung der IT-Projektergebnisse in amerikanischen Unternehmen (vgl. Johnson, 1999, 2002)

Tabelle 2.1 zeigt, dass sich der Anteil erfolgreicher Projekte im Zeitraum zwischen 1994 und 2000 um 12% auf 28% gesteigert hat. Aus Johnson (vgl. Johnson, 2007) geht eine weitere Steigerung auf 35% im Zeitraum zwischen 2000 und 2007 hervor und Buschermöhle et al. (vgl. Buschermöhle et al., 2006, S. 244) ermittelten bei ihrer Befragung von 378 deutschen Unternehmen sogar einen Erfolgswert von 50,8%.

Es ist anzunehmen, dass ein wesentlicher Teil der positiven Erfolgsentwicklung auf Lerneffekte und verbessertes Projektmanagement zurückzuführen ist. Aus Buschermöhle et al. (vgl. Buschermöhle et al., 2006, S. 224) geht

³ unter Dead End ist eine Entwicklungssackgasse zu verstehen, bei der es keine Möglichkeit mehr gibt, das ursprüngliche Projektziel zu erreichen

⁴ befragt wurden 365 amerikanische IT-Verantwortliche

hervor, dass neben den klassischen Managementaktivitäten Know-how und Motivation des Projektteams weitere wichtige Erfolgsfaktoren sind.

Projektmanagement ist methodische Grundfertigkeit für jedes IT-Projekt und muss daher bereits bei der Entscheidungsvorbereitung gefordert werden.

2.1.3 Konsequenzen grundlegender Begriffe des Einsatzes von Anwendungssystemen

Zunächst wurde der Begriff „*Anwendungssystem*“ definiert, „*Prozess*“ und „*Workflow*“ gegeneinander abgegrenzt und der Begriff „*Qualität*“ erläutert (siehe Abschnitt 2.1.1). Ein Zusammenhang zwischen diesen Begriffen und deren elementare Bedeutung für diese Arbeit wird anhand der Tatsache deutlich, dass ein entscheidendes Ziel des Einsatzes von Anwendungssystemen die effiziente Unterstützung betrieblicher Prozesse und Arbeitsabläufe ist (vgl. Mertens et al., 2004, S. 3 f.).

Die Eignung von Anwendungssystemen hinsichtlich der Erreichbarkeit ökonomischer, technischer oder strategischer Ziele wird als deren Qualität (siehe Abschnitt 2.1.1) bezeichnet. Qualität ist damit entscheidungsrelevant.

Während der gesamten Entscheidungsvorbereitung ist Projektmanagement grundlegendes Werkzeug zur Beherrschung von Komplexität, Handhabung von Risiken und Sicherstellung von Qualität, wie in Abschnitt 2.1.2 dargestellt wird. Auf eine detaillierte Darstellung von Projektmanagement und dessen Teildisziplinen wird im Hinblick auf den Umfang dieser Arbeit verzichtet.

2.2 Strategische Faktoren des Einsatzes von OSS

Im Rahmen der strategischen Ausrichtung der Informationsinfrastruktur und damit letztlich auch bei der Beurteilung von Anwendungssystemalternativen sprechen eine Reihe von Faktoren für den Einsatz quelloffener Software (siehe Abschnitt 2.2.1). Daneben kann die häufig modulare Gestaltung von OSS Unternehmen helfen, den von gesetzlichen Regelungen häufig geforderten Nachweis über die Funktionsweise ihrer Informationssysteme zu erbringen und/oder rechtskonforme Anwendungssysteme zu implementieren (siehe Abschnitt 2.2.3).

Allerdings gibt es auch Nachteile beim Einsatz von OSS. Eine Reihe rechtlicher Implikationen (siehe Abschnitt 2.2.4) sind mit der Nutzung und vor allem der etwaigen Weiterentwicklung und Vermarktung von OSS verbunden und ergeben sich aus den speziellen OSS-Lizenzen. Die Suche und Auswahl geeigneter Servicepartner ist je nach Geschäftsmodell der am Markt vertretenen Anbieter schwierig (siehe Abschnitt 2.2.2). Die zeitweise oder auch bewusst dauerhafte Heterogenität der Infrastruktur stellt darüber hinaus ein technisches und betriebswirtschaftliches Risiko dar. Steigende Aufwände für Mitarbeiterschulung und höhere Anforderungen an Know-how und Projektmanagement (siehe Abschnitt 2.1.2) sind weitere strategische Nachteile, die sich allerdings recht gut monetär bewerten lassen und somit bei der ökonomischen Bewertung berücksichtigt werden.

Bei der Auswahl und Bewertung von Anwendungssystemalternativen müssen also neben den technischen und ökonomischen Grundlagen (siehe Abschnitt 2.3, 2.4) die strategischen Faktoren bekannt sein, da eine fundierte Entscheidung nur unter deren Berücksichtigung getroffen werden kann. Gerade langfristige strategische Faktoren (z. B. Vermeidung von Anbieterabhängigkeiten) sprechen dabei häufig für den Einsatz quelloffener Software.

2.2.1 Strategische Betrachtung von OSS

Die komplette strategische Ausrichtung auf OSS ist selten möglich, da es üblicherweise nicht für jede Systemkomponente geeignete OSS gibt. Der Umgang mit neuen Technologien birgt neben rein ökonomischen immer auch technische Risiken (vgl. Lutz et al., 2004, S. 21), die eine Einführung zunächst teilweise und/oder im kleinen Umfang sinnvoll erscheinen lassen. Darüber hinaus wird ein großer Teil von Software nach wie vor als Individualsoftware (siehe Abschnitt 2.3.3) eigens für das jeweilige Unternehmen und dessen Bedürfnisse entwickelt, auch wenn dies immer häufiger auf Basis oder mit Hilfe von Open Source Software geschieht.

Unter strategischen Gesichtspunkten bietet der Einsatz von OSS dennoch Vorteile, die über reine Kostenfaktoren hinaus gehen. Viele dieser Vorteile freier Software, wie z. B. Anbieterunabhängigkeit, Interoperabilität und Erweiterbarkeit, resultieren direkt oder indirekt aus dem offenen Entwicklungsprozess der Software und deren speziellen Lizenzen.

Bei der strategischen Betrachtung wird oft die Anbieterabhängigkeit beim Einsatz proprietärer Software als wesentlicher Nachteil empfunden. Die große Marktkonzentration und die damit verbundene Anbieterabhängigkeit haben dafür gesorgt, dass Softwareanbieter Kundenwünsche und Kundenbedürfnisse weitgehend aus dem Auge verloren haben.

Beim offenen Entwicklungsprozess steht das Anwenderinteresse im Mittelpunkt der Softwareentwicklung, da fehlende Funktionalität vergleichsweise einfach hinzuentwickelt werden kann. Die häufig räumlich verteilte Entwicklung sorgt zum einen für einen strukturierteren Quellcode und damit für leichte Erweiterbarkeit, zum anderen aber auch für Integrierbarkeit. Durch den offenen Quellcode sind nicht ausreichende Schnittstellen schnell erweitert oder neu entwickelt. Die Tatsache, dass OSS häufig gegen etablierte CSS konkurrieren muss, sorgt darüber hinaus dafür, dass OSS tendenziell viele Schnittstellen besitzt.

Da proprietäre Protokolle und Formate bei freier Software ein Widerspruch in sich wären, entstehen offene Standards und sorgen so für Interoperabilität von OSS (vgl. Lutz et al., 2004, S. 21). Ebenfalls durch den konföderierten Entwicklungsprozess wird begünstigt, dass Open Source Software häufig als tendenziell sicherer eingestuft wird. Auch wenn diese These schwer belegbar ist und Anbieter proprietärer Software häufig anmerken oder konstatieren, Sicherheit entstünde auch durch das Verbergen von Sicherheitslücken⁵, sprechen sowohl die meisten Experten als auch viele Belege für die Gültigkeit dieser These (vgl. Renner et al., 2005, S. 171).

2.2.2 Open Source Geschäftsmodelle

Entgeltliche Nutzungslizenzen stehen prinzipiell im Widerspruch zu den Statuten der Open Source Bewegung. Es gibt allerdings Open Source Anbieter, die für ihr Produkt auch eine kommerzielle Lizenz anbieten, um darauf basierende kommerzielle Entwicklungen zu ermöglichen und davon selbst zu profitieren (dieses „Dual Licensing“ machte vor allem die Firma MySQL AB salonfähig). Dieses Vorgehen ist allerdings prinzipiell nur dann möglich, wenn der zu Grunde liegende Code gänzlich Eigentum des Anbieters ist oder für fremde Bestandteile wiederum kommerzielle Lizenzen erworben wurden. Es ist daher nicht weit verbreitet.

Die Tatsache, dass immer mehr Anbieter proprietärer Anwendungssoftware wie SAP und große Hardwarehersteller wie IBM und zuletzt auch DELL eine linuxbasierte Plattform explizit unterstützen bzw. anbieten, ist mitverantwortlich für den Erfolg, den Distributoren verzeichnen. Durch den stark gestiegenen Einsatz von Linux in Unternehmen entstand in der Vergangenheit ebenfalls eine breite Nachfrage nach Support und Serviceleistung für OSS in diesem Bereich. Da die Verfügbarkeit von Know-how, Service und Support wichtiger strategischer Faktor für die Alternativenauswahl und de-

⁵ auch als „Security by obscurity“ bekannt

ren Bewertung ist, werden die wichtigsten Geschäftsmodelle im Folgenden erklärt.

Distributor

Das Geschäftsmodell der Distributoren besteht im Wesentlichen aus drei Wertschöpfungselementen.

1. Entgeltliche Bereit- und Zusammenstellung der Software auf Datenträgern (i.d.R. CD's oder DVD's).
2. Einfach zugänglicher Supportservice
3. Garantierte Bereitstellung von Softwareupdates

Vor allem Unternehmen haben ein großes Interesse an der Verfügbarkeit von Datenträgern. Die schnelle Verfügbarkeit von Support und individueller Problemlösung ist in vielen Fällen ebenso geschäftskritisch wie die Verfügbarkeit von Updates. Außerdem ist die Nutzung einer zertifizierten und durch einen der großen Linuxdistributoren vertriebenen Version des Betriebssystems oft eine geforderte infrastrukturelle Basisausstattung proprietärer Anwendungssoftwareanbieter wie z.B. SAP.

Mit Novell (Suse Linux) und RedHat gibt es zwei Anbieter auf dem Markt, die zertifizierte Linuxplattformen vertreiben. RedHat war 1999 das erste Unternehmen im Open Source Umfeld, das den Schritt an die Börse vollzog (vgl. Shankland, 1999).

Entwickler

Eine Reihe von Dienstleistern entwickeln Open Source Software weiter. Dabei ist zu unterscheiden zwischen jenen, deren Basis einer GPL-ähnlichen

Lizenz unterliegt und denen, deren Basis die Aufnahme der Open Source Software in proprietären Code erlaubt (fehlendes oder schwaches Copyleft). Während letztere das abgeleitete Produkt nicht zwangsläufig wieder unter eine Open Source Lizenz stellen müssen und so direkt Erträge aus dem Vertrieb und dem Support der Software erzielen können, profitieren erstere vorwiegend von der schneller wachsenden Nutzerbasis und Verbreitung der Software. Ihre Wertschöpfung besteht vorwiegend aus bezahlten Entwicklungsprojekten nach Unternehmensauftrag oder Entwicklungsprojekten mit Kostenteilung unter mehreren Interessenten (sogenanntes Shared-Finance Modell).

Service Provider

Für Service Provider spielt es keine Rolle, welche OSS genau eingesetzt wird oder woher sie stammt. Der Service wird in einem breiten Bereich von OSS angeboten, solange bestimmte Kriterien an die Software erfüllt sind (vgl. Krishnamurthy, 2005, S. 285). Häufig ist dabei die Tatsache entscheidend, dass sich die Service Provider in regionaler Nähe des Kunden befinden (vgl. Meyer und Brunke, 2005, S. 54). Krishnamurthy (vgl. Krishnamurthy, 2005, S. 286) stellt dar, dass die Wirksamkeit dieses Geschäftsmodells wesentlich davon abhängt, ob es dem Service Provider gelingt, der OSS einen wirklichen Mehrwert hinzuzufügen. Dies kann zum Beispiel durch die Integration verschiedener OSS zu einem Produktbundle oder lediglich durch die Auswahl der für einen Kunden geeigneten OSS geschehen. Laut Leiteritz (vgl. Leiteritz, 2004, S. 139 ff.) ist Dienstleistung im Umfeld von OSS das einzige nachhaltig erfolgreiche Geschäftsmodell. Das wird durch die Tatsache gestützt, dass kaum ein Marktteilnehmer im Bereich Open Source auf ein Dienstleistungsangebot verzichtet. Eine Transformation anderer Geschäftsmodelle hin zu diesem ist erkennbar. Die Aufnahme von Servicepartnern in das Informationsangebot der Plattform Sourceforge ist dafür ein Beispiel.

2.2.3 Rechtsgrundlagen des Einsatzes von OSS

Verursacht durch den andauernden Patentrechtsstreit zwischen SCO und IBM (vgl. Heise, 2003) sind vor allem potentielle Nutzer im Hinblick auf die Rechtmäßigkeit der Nutzung von Linux sowie die Wirksamkeit von Open Source Lizenzen verunsichert. Die Patentierbarkeit von Computerprogrammen ist zentraler Bestandteil der aktuellen juristischen Fachdiskussion. In diesem Zusammenhang sorgten der vorerst verworfene Gesetzentwurf der „*Richtlinie über die Patentierbarkeit computerimplementierter Erfindungen*“, welcher erstmalig die Patentierbarkeit von Software auch im europäischen Rechtsraum regeln soll, sowie die Entwürfe zur GPL Version 3 für kontroverse Diskussionen zwischen den beteiligten Interessengruppen⁶.

Insbesondere die leichte Nachweisbarkeit von Patentverletzungen bei offenliegendem Sourcecode sorgt für eine Informationsasymmetrie, welche die Open Source Entwicklung nachhaltig negativ beeinflussen kann. Der Verzicht auf Geltendmachung von Ansprüchen aus der Urheberschaft ist ein zentraler Bestandteil von Open Source Lizenzen. Ein weitreichender Patentschutz stünde dazu im Widerspruch. Außerdem scheint ein staatlich garantiertes Monopol, in Anbetracht der kurzen Innovationszyklen der Softwareindustrie, mit 20 Jahren als zu lang bemessen (vgl. Böcker, 2007, S. 515). Insgesamt kann das eigentlich innovationsfördernde Patentrecht – falsch gestaltet – die zukünftige Entwicklung der Open Source Bewegung stark beeinträchtigen. Insbesondere die Breite eines künftigen Patentschutzes und die Auslegbarkeit „erfinderischer Leistung“ muss daher in einer IT-Strategie Beachtung finden.

Im Gegensatz zu „Public Domain Software“, wo die Urheberrechte vollständig an die Allgemeinheit abgetreten werden⁷, regelt eine OSS-Lizenz die Ausübung des Urheberrechts, indem Anwendern ein einfaches Nutzungsrecht gemäß §31 Abs. 2 UrhG eingeräumt wird (vgl. Koglin und Metzger, 2004, S. 295). Darüber hinaus regelt eine OSS-Lizenz üblicherweise die Ausübung vier

⁶ letztere wurde vor kurzem veröffentlicht

⁷ ist im europäischen Rechtsraum nicht möglich (vgl. Renner et al., 2005, S. 14))

zentraler Freiheiten, die im Folgenden beschrieben werden. Gleichzeitig werden den Nutzern in Abhängigkeit vom Lizenztyp gewisse Verpflichtungen auferlegt. Dies betrifft vor allem die Veröffentlichung etwaiger Änderungen, die Zugänglichmachung des Quellcodes und die Weiterverbreitung (Distribution) der geänderten OSS.

Copyleft und freie Software

„Copyleft“ ist eine bewusste Modifikation des Begriffes „Copyright“. Eine Copyleft-Lizenz verlangt, dass alle modifizierten und erweiterten Versionen einer mit diesem Konstrukt lizenzierten Software wieder derselben Lizenz unterliegen (dies wird auch als „*viraler Effekt*“ bezeichnet) (vgl. Wichmann, 2004, S. 7). „*Freie Software*“ nach Definition der Free Software Foundation (FSF) gewährt dem Nutzer vier zentrale Freiheiten (vgl. Stallman, 2007):

1. Die Freiheit, die Software für jeden Zweck zu nutzen.
2. Die Freiheit, die Software zu untersuchen und anzupassen.
3. Die Freiheit, die Software zu kopieren.
4. Die Freiheit, Kopien der Software weiterzugeben.

Den freien Zugang zum Quellcode gewähren alle OSS-Lizenzen. OSS-Lizenzen lassen sich grundsätzlich in zwei Gruppen unterteilen. Lizenzen die ein „Copyleft“-Konstrukt nutzen und Lizenzen ohne dieses Konstrukt. Die häufigste OSS-Lizenz General Public License (GPL) nutzt ein starkes Copyleft. Lizenzen ohne Copyleft (z.B. die BSD-Lizenz) oder mit schwachem Copyleft (z.B. Lesser General Public License (LGPL)) verlangen oft nur einen Copyrightvermerk und ermöglichen so die Nutzung bei kommerzieller Softwareentwicklung.

IT-Compliance

Zahlreiche Richtlinien, Normen und Gesetze (u.a. KonTraG, HIPAA, Basel II, SOX) sorgten in der Vergangenheit für neue oder geänderte Anforderungen an Informationssysteme. Die genannten Richtlinien und Gesetze haben dabei u.a. die Nutzung der IT zur Verbesserung der Geschäftstransparenz zum Zweck. Besonders seitdem der „*Sarbanes-Oxley Act*“ (SOX)⁸ in den USA und die Eigenkapitalrichtlinie „*Basel II*“ in Europa geltende gesetzliche Vorschriften sind, werden viele IT-Projekte mit dem Ziel der Verbesserung oder Erfüllung gesetzlicher Regelungen (engl. Compliance) aufgelegt.

Durch Basel II wird der Umgang mit inhärenten Kreditvergeberisiken geregelt. Es werden in diesem Zusammenhang belegbare Sicherheit und Maßnahmen zur Minimierung von Ausfallrisiken im Kontext allgemein gestiegener Bedeutung von Informationssystemen gefordert (vgl. Haar, 2006).

Mit SOX wird eine Bewertung der Effektivität von internen Kontrollstrukturen eines Unternehmens und der Prozeduren zur Sicherstellung der Persistenz von Daten durch die Unternehmensleitung gefordert und die weitreichende Haftung der Verantwortlichen für die Verletzung dieser Auflagen geregelt (vgl. Norris und Young, 2005, S. 2).

Daneben existieren weitere branchenspezifische oder -übergreifende gesetzliche Vorgaben. Vorgaben zum Datenschutz beinhalten zum Beispiel §9 und §11 des Bundesdatenschutzgesetzes (BDSG), Richtlinien zur Archivierung elektronischer Dokumente die „*Grundsätze ordnungsgemäßer Buchführung*“ (GoB), die „*Grundsätze ordnungsgemäßer DV-gestützter Buchführungssysteme*“ (GoBS), die „*Grundsätze zum Datenzugriff und zur Prüfbarkeit digitaler Unterlagen*“ (GDPdU) sowie Regelungen zur Informationspflicht elektronischer Dienste das Telemediengesetz TMG (vgl. Rath, 2007).

⁸ (vgl. Sarbanes und Oxley, 2002)

IT-Compliance dient dabei keinem Selbstzweck. Die Verletzung von Gesetzen und Richtlinien kann erhebliche wirtschaftliche, zivil- und strafrechtliche Folgen haben. Nach Rath (vgl. Rath, 2007) können dabei auch Kosteneinsparpotentiale, zum Beispiel durch die Einführung von Serviceorientierten Architekturen (siehe Abschnitte 2.3.3), erschlossen werden.

2.2.4 Konsequenzen strategischer Faktoren des Einsatzes von OSS

Bei der Entscheidungsvorbereitung sind drei wesentliche strategische Überlegungen anzustellen.

1. Zunächst muss geklärt werden, ob direkte strategische Vor- und Nachteile von Open Source Software für den speziellen Anwendungsfall relevant sind.
2. Weiter ist die Notwendigkeit, Verfügbarkeit, sowie ggfs. Auswahl des richtigen Dienstleisters, Entwicklers und/oder Distributors eine strategische Frage, da alternativ oder begleitend auch in den Aufbau eigenen Fachwissens investiert werden kann.
3. Die besonderen rechtlichen Rahmenbedingungen von Open Source Lizenzen und deren nicht abschließend geklärte Gültigkeit sowie die kaum vorhersagbare Entwicklung müssen ebenso Bestandteil eines Strategiekalküls sein wie die Frage, ob auf Basis von Open Source Software zum Zeitpunkt der Entscheidung ebenso wie für die Zukunft rechtskonforme Anwendungssysteme darstellbar sind bzw. sein werden.

Die Analyse des Marktes an Distributoren, Entwicklern und Service Providern ist wichtig, wenn ein Unternehmen OSS lediglich einsetzen will. In diesen Fällen erlangt die Organisation zwar mit der Zeit Anwendungswissen, das sie im weiteren Verlauf auch für die Schulung und Unterstützung

neuer Anwender einsetzen kann, dennoch ist sie zur Klärung spezieller Fragen oder zeitkritischen Behebung von Fehlern auf Fach- und/oder Entwicklungswissen angewiesen. Diesen Bereich decken entsprechende Dienstleister ab. Die Auswahlkriterien für den richtigen Dienstleister unterscheiden sich dabei nicht wesentlich von der Auswahl im Bereich proprietärer Software. Ein zweckmäßiges Vorgehen ist zum Beispiel, Beratungsleistung direkt von entsprechend organisierten Entwicklern zu beziehen. Auch Distributoren bieten häufig entsprechenden Service an. Bei einem geplanten Einsatz umfangreicher OSS-Anwendungssysteme aus vielen Einzelkomponenten bieten sich auch Service Provider an, welche mit Abschluss von Wartungsverträgen den Service und Support für das gesamte Anwendungssystem sicherstellen und ggfs. sogar die Weiterentwicklung der Systemkomponenten durchführen oder zumindest mit weiteren Dienstleistern koordinieren.

Die Analyse der eigenen rechtlichen Situation im Umgang mit OSS ist ebenfalls strategisch relevant. Bereits durch den Erwerb einer rechtmäßig in Umlauf gebrachten Kopie einer Software, erlangt der Anwender die Berechtigung zur bestimmungsgemäßen Nutzung (vgl. Langmack, 2006, S. 374). Bei OSS-Einführungsprojekten ist daher zunächst zu prüfen, ob über die bestimmungsgemäße Nutzung im Sinne des §69d UrhG hinaus vom erweiterten Nutzungsrecht an der Software Gebrauch gemacht werden soll⁹, da nur in diesem Fall ein Lizenzvertrag mit den Urhebern abgeschlossen wird. Nur dann sind die in Open Source Lizenzen geregelten Pflichten (siehe Abschnitt 2.2.3) überhaupt praktisch relevant (vgl. Langmack, 2006; Widmer und Baehler, 2006, S. 378, S. 174). Strittig ist allerdings, ob die innerbetriebliche Weitergabe von OSS über Abteilungsgrenzen hinweg bereits eine Distribution im Sinne der erweiterten Nutzungslizenz darstellt. Die aktuelle Rechtsprechung geht von der Bejahung dieses Sachverhaltes aus, wonach praktisch jedes Unternehmen von der erweiterten Nutzungslizenz Gebrauch macht und damit die zugrundeliegende OSS-Lizenz implizit annimmt.

⁹ Das Anlegen von Sicherheitskopien (§69d Abs. 2 UrhG), Fehlerberichtigung (§69d Abs. 1 UrhG) und Dekompilierung zur Herstellung der Interoperabilität (§69e UrhG) fallen dabei unter den Terminus „bestimmungsgemäße Nutzung“

Für Unternehmen, die OSS nutzen, ist insbesondere die Haftungs- und Gewährleistungssituation bedeutend. In gängigen OSS-Lizenzen werden sowohl Haftung als auch Gewährleistung generell ausgeschlossen. Dieser generelle Ausschluss ist in europäischen Rechtssystemen i. d. R. nicht wirksam. In Fällen, in denen die Software nicht durch Abschluss eines Kaufvertrages von einem Distributor bezogen wird, sondern z. B. aus dem Internet heruntergeladen wurde, handelt es sich rechtlich um eine Schenkung. Das deutsche Schenkungsrecht begrenzt die Gewährleistung auf arglistige Täuschung und die Haftung auf grobe Fahrlässigkeit sowie Vorsatz (vgl. Renner et al., 2005, S. 23). Diesem Faktor ist bei strategischen Betrachtungen Rechnung zu tragen, da sich zum Teil nicht unerhebliche Haftungsfolgen für Nutzer (z. B. Preisgabe personenbezogener Daten) mangelhafter Software ergeben können. Dies zeigt erneut deutlich, dass Qualität im Allgemeinen (siehe Abschnitt 2.1.1) - und Softwarequalität (siehe Abschnitt 2.3.4) im Besonderen - beurteilt werden muss und damit entscheidungsrelevant ist.

Wird OSS als Basis zur Weiterentwicklung genutzt, sind die Lizenzen der Ausgangsprodukte auf Kombinierbarkeit zu untersuchen, insbesondere wenn mehrere OSS-Produkte zu integrieren sind. Auch sind die rechtlichen Verpflichtungen, die eine Lizenz beinhaltet, auf das Ziel der geplanten Weiterentwicklung hin zu prüfen. Bei geplantem Vertrieb von Software sind darüber hinaus die Bestimmungen zur Kenntlichmachung der Urheberschaft und Zugänglichmachung des Sourcecodes (falls die OSS im kompilierten Objektformat vertrieben wird) zu prüfen und ein Lizenzierungstyp zu bestimmen (falls die Ausgangsprodukte keiner Copyleftlizenz unterliegen) (vgl. Widmer und Baehler, 2006, S. 174 ff.).

Zuletzt ist die Eignung einer Anwendungssystemalternative hinsichtlich der Etablierung und Sicherstellung unternehmenseigener oder staatlich vorgegebener Richtlinien, Gesetze und Normen zu prüfen. Wie bereits beschrieben kann Open Source Software wegen ihres modularen Aufbaus in diesem Bereich zweckdienlich sein.

2.3 Technische Grundlagen von Anwendungssystemen

Im folgenden Abschnitt werden zunächst Migration und Substitution voneinander abgegrenzt und erläutert, Hardwarearchitekturen vorgestellt und die systematische Klassifizierung von Software erläutert. Im Anschluss wird der in Abschnitt 2.1.1 eingeführte Qualitätsbegriff im Kontext technischer Anwendungssystemeigenschaften erläutert. Den Abschluss bildet die Darstellung der Konsequenzen für das entwickelte Entscheidungsmodell.

2.3.1 Substitution / Migration

Die Art der geplanten Einführung eines Anwendungssystems ist entscheidend für deren Komplexität. Der einfachste Fall ist die Umstellung eines vollkommen manuell ablaufenden Geschäftsprozesses auf teil- oder vollautomatisierte Abwicklung durch ein Anwendungssystem. Da keine Altsysteme oder Altdaten übernommen werden müssen und Anwender auch nicht von den Vorteilen eines „neuen“ Anwendungssystems gegenüber einem Altsystem überzeugt werden müssen, reduziert sich die Komplexität der Einführung des AS erheblich.

Unter Substitution ist ähnlich wie im vorher genannten Fall die Einführung eines Anwendungssystems zu verstehen. Allerdings stellt es dabei den Ersatz für ein bereits bestehendes System dar. Dessen Daten und Workflows werden dabei nicht übernommen. Kosten für die Durchführung und Planung einer Datenmigration entfallen und reduzieren damit den Aufwand und die Komplexität des Projektes. Allerdings ist insgesamt mehr Schulungsaufwand zu betreiben, wenn ein erheblicher Teil des Aufwands auf das „Entlernen“ geübter Arbeitsabläufe entfällt.

Die Migration ist der häufigste Fall der Einführung eines neuen Anwendungssystems. Im Vergleich mit den ersten beiden Fällen muss ein erhöhter Schulungsaufwand berücksichtigt werden, weil bei Übernahme von Daten und Arbeitsabläufen üblicherweise strukturelle Änderungen gemacht werden müssen oder bewusst gemacht werden (vgl. Eisendle, 2001, S. 306). Darüber hinaus fordert der häufig notwendige zeitweise parallele Betrieb des alten und des neuen Anwendungssystems („*weiche Migration*“) eine besonders aufmerksame Planung. Im Gegensatz zur „*weichen / sanften*“ Migration wird unter „*harter / schneller*“ Migration das gleichzeitige Abschalten des Alt- und Anschalten des Neusystems verstanden (vgl. Hoegner et al., 2003; Langmack, 2006, S. 11, S. 490 f.). Praktisch zu jeder Zeit wird während einer Migration mit Produktivdaten gearbeitet, so dass erhöhter Aufwand durch die Sicherstellung von Verfügbarkeit und Datenintegrität entsteht. Es wird zwischen „*ablösender*“ und „*fortführender*“ Migration unterschieden (vgl. Langmack, 2006, S. 21). Unter ablösender Migration wird dabei der Wechsel eines Softwareanbieters verstanden, also zum Beispiel die Migration einer Oracle Datenbank auf MySQL. Unter fortführender Migration wird im Gegensatz dazu die Beibehaltung eines Softwareherstellers verstanden, also z. B. Windows XP auf Windows Vista (vgl. Langmack, 2006, S. 21). Bei einer fortführenden Migration können gewohnte Werkzeuge und Arbeitsschritte teilweise erhalten bleiben. Entsprechende anbieterspezifische Migrationshilfen erleichtern die Daten- und Softwaremigration.

2.3.2 Hardware

Die Befürchtung, das Moor'sche Gesetz (vgl. Moore, 1965) zur Voraussage der Entwicklung integrierter Schaltkreise besäße angesichts unüberwindbarer technischer Grenzen bald keine Gültigkeit mehr, hat sich angesichts der aktuellen Entwicklung im Hardwaresektor nicht bewahrheitet (vgl. Heise, 2005).

Die beiden größten Anbieter von x86-Prozessortechnologie, Intel und AMD, überbieten sich regelmäßig mit Ankündigungen und neuen Technologien¹⁰.

Während im Supercomputersegment¹¹ Clusterserver der x86-Architektur längst dominieren (vgl. Meuer, 2006, S. 17), werden immer mehr Hostsysteme und Mainframes aus den Midsize Marktsegmenten ebenfalls durch Clusterserver aus x86- bzw. fortentwickelten AMD64- / IA64- / EM64T-Architekturen ersetzt. Die Reduzierung der Wartungs- und Pflegekosten sprechen laut Wichmann (vgl. Wichmann, 2004, S. 37) insbesondere bei Neuanschaffungen für die Clusterservertechnologie.

Oft sind Migrationsprojekte mit Serverkonsolidierungen verbunden oder sogar Grund für diese Projekte. Sogenannte Bladeserver gelten dabei als das bauartliche Zukunftskonzept. Sie ermöglichen sehr hohe Rechenleistungsdichten und Virtualisierung bei ebenfalls guter Skalierbarkeit (vgl. Klippstätter, 2004). Daneben besteht prinzipiell auch die Möglichkeit der Konsolidierung auf neue Mainframes.

Insgesamt ist ein Trend hin zur Standardisierung von Hardware zu beobachten. Als Verbindungsstandard in der Breite setzt sich immer mehr Fast-Ethernet bzw. dessen Nachfolger Gigabit-Ethernet durch, während sich im Hochleistungsbereich Infiniband durchzusetzen scheint (vgl. Waldman, 2006, S. 4 f.). Auch verteilte Speichersysteme mit Netzanbindung, sogenannten Storage Area Networks (SAN), werden immer häufiger in Unternehmensrechenzentren genutzt.

Die Treiber grundlegender Hardwarekomponenten sind entweder im Linuxkernel enthalten oder als Kernelmodul verfügbar. Periphere Hardware hingegen wird teilweise schlecht oder gar nicht unterstützt. Die Unterstützung muss deshalb in der Entscheidungsphase genau geprüft werden. In manchen Fällen stellen die Hardwareanbieter nur proprietäre Treiber und auch nur für bestimmte Betriebssystemversionen zur Verfügung. Im Einzelfall kann dies

¹⁰ Ein guter Marktüberblick findet sich bei Windeck (vgl. Windeck, 2007, S. 164 ff.)

¹¹ (vgl. Stahlknecht und Hasenkamp, 2005, S. 15)

dazu führen, dass neue Hardware beschafft werden oder ein Projekt gar als nicht vorteilhaft eingestuft werden muss.

2.3.3 Software

Nach Mertens et al. (vgl. Mertens et al., 2004, S. 21) bildet Software die Voraussetzung für den Betrieb eines Rechners und bezeichnet allgemein in einer Programmiersprache geschriebene Programme, die nach Übersetzung auf einem Rechner ausführbar sind. Dabei wird je nach Nähe zum Betriebssystem zunächst zwischen „*Systemsoftware*“ und „*Anwendungssoftware*“ unterschieden. Weiter wird Anwendungssoftware wieder in Standardsoftware und Individualsoftware unterteilt (siehe Abbildung 2.2).

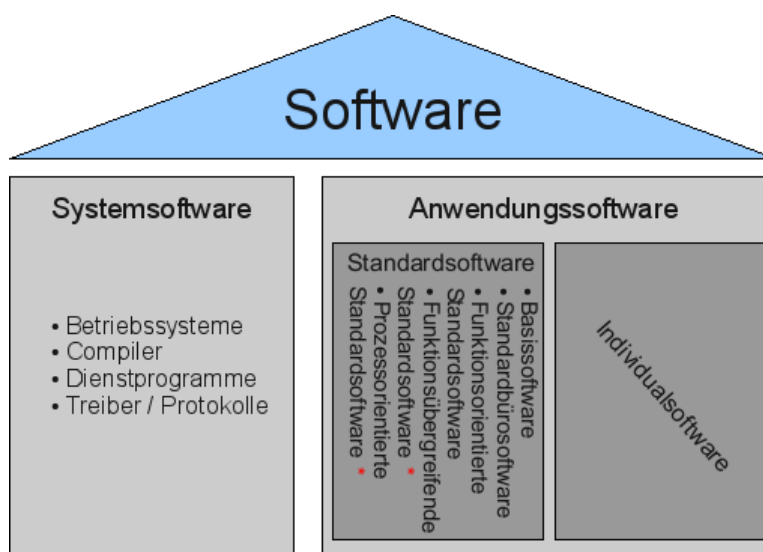


Abbildung 2.2: Eigene Darstellung in Anlehnung an Mertens et al. (vgl. Mertens et al., 2004, S. 21) - Klassifikation von Software

Systemsoftware umfasst dabei neben dem Betriebssystem auch Treibersoftware zur Kommunikation mit Peripherie und anderen Computern im Netzwerk sowie häufig benötigte Programme, etwa zur Dateiverwaltung. Basissoftware implementiert Funktionen, die unabhängig vom spezifischen An-

wendungsgebiet benötigt werden (E-Mail, Browser, Serverdienste, Virencanner). Standardsoftware wird je nach ihrem Anwendungsgebiet in Standardbürosoftware (MS-Office, Open Office, StarOffice etc.), funktionsorientierte Standardsoftware für eine bestimmte betriebliche Funktion (z. B. AutoCAD), funktionsübergreifende Standardsoftware (z. B. SAP R/3) und prozessorientierte Standardsoftware (z. B. Workflow Management Systeme) unterteilt, wobei die Grenzen fließend sind (vgl. Mertens et al., 2004, S. 32). Individualsoftware ist für die Bedürfnisse eines einzelnen Unternehmens speziell entwickelte Software. Abbildung 2.2 zeigt die Klassifikation von Software grafisch und wurde um „funktionsübergreifende“ und „prozessorientierte“ Standardsoftware gemäß Mertens et al. (vgl. Mertens et al., 2004, S. 32) erweitert.

Auch die Grenzen zwischen Standard- und Individualsoftware schwinden mehr und mehr. Im Zuge fortschreitender Modularisierung und Entwicklung komponentenbasierter Anwendungssysteme (z. B. J2EE/Eclipse, .NET) wird Individualsoftware immer häufiger aus Standardkomponenten zusammengestellt. Durch die Verbreitung des Internets und dessen Basistechnologien (TCP/IP, HTTP, SMTP, XML) wurde das Konzept der unternehmensweiten Anwendungsintegration „*Enterprise Application Integration*“ (EAI) durch den umfassenderen Ansatz „*Serviceorientierter Architekturen*“ (SOA) abgelöst bzw. erweitert (vgl. Mertens et al., 2004; Jakob, 2006, S. 33 f.). Befasst sich EAI im Wesentlichen mit der Integration verschiedene Anwendungsdomänen umfassender Standardsoftware (sogenannter Siloapplikationen) innerhalb eines Unternehmens mittels häufig proprietärer Integrationssoftware, erweitert SOA diesen Ansatz im Zuge steigenden Kosten- und Standardisierungsdrucks um konzeptionell plattform- und anbieterunabhängige Integration (häufig mittels Webservices / SOAP). Die Aufgabe zur Etablierung einer Anwendungssysteminfrastruktur in einer SOA-Umgebung besteht dabei ähnlich wie die Aufführung eines Musikstücks aus dessen Komposition, Orchestrierung und Choreografie (vgl. Thilloy, 2006, S. 5 f.). Die Integration externer Services mittels „*Application Service Providing*“ (ASP) wird dadurch ebenfalls möglich. SOA ist, anders als der stark tech-

nikgetriebene EAI Ansatz, ein umfassendes Managementkonzept, das eine technische Ausprägung beinhaltet.

2.3.4 Softwarequalität und Reife von OSS

Qualität ist, wie in Abschnitt 2.1.1 dargelegt, Bestandteil fundierter Entscheidungen zum Einsatz von Anwendungssystemen. Softwarequalität wird dabei in Erweiterung der allgemeinen Qualitätsdefinition nach Balzert (vgl. Balzert, 1997, S. 257) als „Gesamtheit der Merkmale und Merkmalswerte eines Softwareprodukts, die sich auf dessen Eignung beziehen, festgelegte oder vorausgesetzte Erfordernisse zu erfüllen“, definiert.

Die Normen ISO/IEC 9126 und DIN 66272 legen sechs Qualitätsmerkmale fest:

1. Funktionalität
2. Zuverlässigkeit
3. Benutzbarkeit
4. Effizienz
5. Änderbarkeit
6. Übertragbarkeit

Diese sechs Hauptmerkmale werden durch entsprechende Teilmerkmale genauer spezifiziert. Beide Normen stellen eine Grundlage zur Erstellung sogenannter FCM-Qualitätsmodelle für die Messung von Softwarequalität dar. Dabei wird Softwarequalität durch einen Baum oder ein Netz aus Faktoren (factors), Kriterien (criteria) und Metriken (metrics) operationalisiert. Während die DIN 66272 kein konkretes Modell oder Metriken nennt (vgl.

Mellis, 2001, S. 422), existieren für die ISO/IEC 9126 unter ISO/IEC 9126-1 sowohl ein konkretes Modell als auch entsprechende Metriken¹².

Renner et al. (vgl. Renner et al., 2005, S. 31 f.) nutzen in ihrer OSS-Studie das FURPS-Modell (ebenfalls ein FCM-Modell), nach dem die Qualität von Software durch die Eigenschaften Funktionalität, Benutzerfreundlichkeit, Zuverlässigkeit, Leistung und Support zu beurteilen ist. Zuverlässigkeit / Sicherheit sind dabei, laut der zugrundeliegenden Studie, die wichtigsten Eigenschaften von Software, gefolgt von Funktionalität, Benutzerfreundlichkeit, Leistung und Support (jeweils im Anwenderurteil).

Viele der vermeintlichen Vorteile (siehe Abbildung 2.3) von OSS wie „Stabilität“, „Sicherheit“, „Interoperabilität“, „Anpassbarkeit“ (vgl. Riklin, 2006; Renner et al., 2005, S. 16 f., S. 16 f.) sind produktbezogene Qualitätsmerkmale und lassen sich durch FCM-Modelle messbar machen. Der OSS-Entwicklungsprozess begünstigt die vergleichsweise positive Ausprägung vieler dieser Qualitätsmerkmale. Generell lässt sich daraus jedoch keine PRO/KONTRA Aussage zu OSS herleiten (vgl. Neumann, 2005, S. 123 ff.).

Während viele Anwender OSS ein hohes Maß an „Sicherheit“ zuschreiben (vgl. Renner et al., 2005, S. 56), sind die „Nutzerfreundlichkeit“ und „Skalierbarkeit / Performance“ von OSS kritische Merkmale. Die Tatsache, dass OSS in der Regel von Spezialisten entwickelt wird, für die sonstige technische Faktoren häufig im Vordergrund stehen, sorgte in der Vergangenheit für einen vergleichsweise schlechten Ruf in Bezug auf die Nutzerfreundlichkeit (vgl. Wichmann, 2004, S. 52). Außerdem ist OSS zunächst oft mit beschränkter Funktionalität entwickelt worden, die einen Einsatz im Hochleistungsbereich (HPC) schwierig machte (vgl. Wichmann, 2004, S.53).

Durch das Engagement großer Konzerne¹³ hat sich dieser Zustand jedoch geändert (vgl. Wichmann, 2004, S. 52). Mittlerweile gibt es viele grafisch und

¹² in ISO/IEC TR 9126-2 bis ISO/IEC 9126-4

¹³ z.B. IBM, Novell, SUN

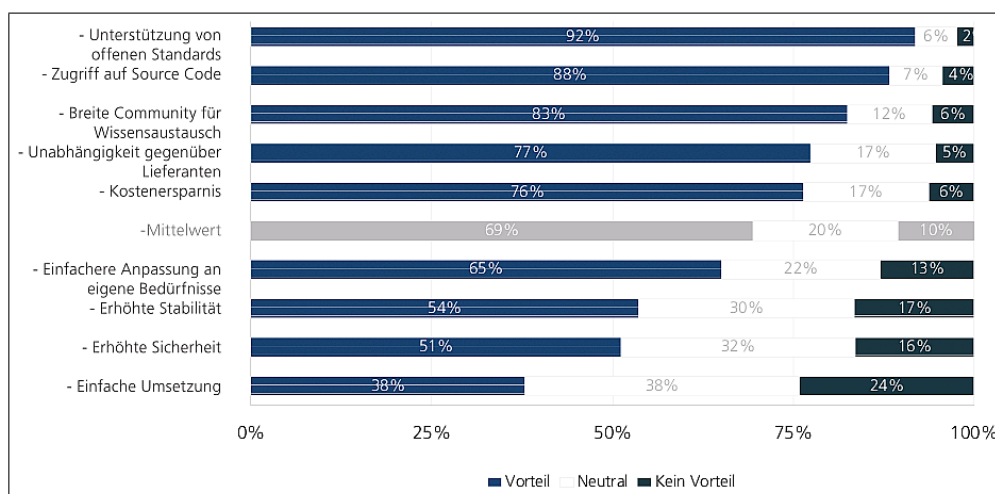


Abbildung 2.3: Vorteile von Open Source Software nach Riklin (vgl. Riklin, 2006, S. 16)

häufig sogar intuitiv nutzbare Benutzerschnittstellen (siehe z.B. Gnome oder KDE). Darüber hinaus gibt es mittlerweile für den Unternehmenseinsatz sehr gut geeignete OSS (z. B. MySQL, JBoss, RedHat Linux), wie der Katalog von Otis et al. (vgl. Otis et al., 2007) zeigt.

In Verbindung mit der Qualität von Software steht auch der Begriff „Reife“. Dieser resultiert, anders als der produktbezogene Qualitätsbegriff, aus dem prozessorientierten Qualitätsansatz und wird i. d. R. mittels sogenannter Reifegradmodelle (Capability Maturity Modell (CMM), Bootstrap, Software Process Improvement and Capability Determination (SPICE) u. a.) gemessen.

Auch in Verbindung mit OSS Software wird der Reifebegriff häufig im Hinblick auf Merkmale des Entwicklungsprozesses operationalisiert. Dies resultiert aus der Schwierigkeit, die produktbezogene Qualität ex ante zu ermitteln, weil dazu i. d. R. aufwendige Produkttests notwendig sind. Wichmann (vgl. Wichmann, 2004, S. 24 ff.) schlägt zu diesem Zweck die Bestimmung der Einsatzreife von OSS wie folgt vor:

Reife des Projekts Men Power, straffe Projektleitung, Alter des Projekts, Anzahl der Releases, Veröffentlichungen

Marktpräsenz der Software Verbreitung bei Endanwendern, Unterstützung durch (kommerzielle) Anbieter, Wettbewerb

Unternehmensorientierung Unterstützung von Unternehmensbedürfnissen, Support und Wartung, Lizenz

Zur Bewertung der einzelnen Merkmale (Reife des Projekts, Marktpräsenz der Software, Unternehmensorientierung) des Modells schlägt Wichmann (vgl. Wichmann, 2004, S. 26) eine Notenskala von 1 (sehr gut) bis 5 (mangelhaft) vor. Dabei werden die Ausprägungen der Indikatoren (z. B. Men Power, Verbreitung bei Endanwendern u. a.) zur Bewertung der Merkmale zusammengefasst. Wichmann (vgl. Wichmann, 2004, S. 26) weist darauf hin, dass eine unternehmensspezifische Gewichtung der Indikatoren vorgenommen werden kann. Auf die konkrete Festlegung von Metriken wird verzichtet. Daher beruht die Festlegung der Indikatorgewichtung und damit die Bewertung der Merkmale auf dem jeweiligen Expertenurteil. Die Indikatorenausprägungen lassen sich auf Basis von Erfahrungen aus vergangenen Projekten empirisch erheben, indirekt durch Websuche ableiten (zum Beispiel projektspezifische Websites / Channels) oder direkt aus Projektdaten (z. B. mittels Sourceforge) ermitteln.

2.3.5 Konsequenzen technischer Grundlagen von Anwendungssystemen

Die Einführung eines neuen Anwendungssystems stellt immer auch eine technische Herausforderung dar. Während sich Hardware vorwiegend hinsichtlich ihrer Rechenleistung weiterentwickelt, Standards wie Ethernet, Infiniband und die x86-basierten Architekturen aber über einen vergleichsweise langen Zeitraum konstant bleiben, verändert sich der Softwaresektor schnell (siehe Abschnitt 2.3.2, Abschnitt 2.3.3). Die Herausforderung besteht darin

– mit möglichst geringen Anforderungen an die Hardware – Software auszuwählen oder zu entwickeln, die gut zu warten, zuverlässig und sicher ist, sowie die funktionalen Anforderungen möglichst vollständig erfüllt. Dabei muss sie möglichst flexibel an zukünftige Standards und Techniken angepasst, erweitert und integriert werden können. Bei der Entscheidung zur Einführung neuer Anwendungssysteme müssen die grundlegenden Anforderungen an die Qualität der verwendeten OSS (siehe Abschnitt 2.3.4) mit Freiheitsgraden, Zukunftsperspektiven und weiteren strategischen Faktoren (siehe Abschnitt 2.2) verglichen und bewertet werden. Die produktbezogene Qualitätsbestimmung von Software ist, wie in 2.3.4 dargestellt, ex ante schwierig. Bei der Alternativenauswahl (siehe Abschnitt 3.3) soll daher zunächst das Reifegradmodell nach Wichmann (vgl. Wichmann, 2004, S. 24 ff.) zum Einsatz kommen. Durch geeignete Pilotprojekte wird später die produktbezogene Qualität (siehe Abschnitt 2.3.4) bestimmt und in die Qualitäts-Bewertung aufgenommen. Ein Zusammenhang zwischen dem Reifebegriff nach Wichmann (vgl. Wichmann, 2004, S. 24 ff.) und der produktbezogenen Softwarequalität (siehe Abschnitt 2.3.4) wird z. B. durch die Erkenntnis deutlich, dass eine Reihe von qualitätsbegünstigenden Faktoren¹⁴ und Möglichkeiten (z. B. Peer Reviews, schnelle Erweiterbarkeit) besondere Merkmale des OSS-Entwicklungsprozesses sind (vgl. Neumann, 2005, S. 123 ff.). Michlmayr (vgl. Michlmayr, 2005) stellt bei einer empirischen Studie erfolgreicher OSS fest, dass effektive Koordinations- und Kommunikationsprozesse eine weitere wichtige Rahmenbedingung auch für die Qualität von OSS darstellen.

¹⁴ u. a. intelligente Nutzung von Compilern und anderen Entwicklungswerkzeugen, ausgeprägte Disziplin unter Entwicklern, anforderungskonforme Entwicklungssysteme und Netzwerkarchitekturen

2.4 Ökonomische Bewertung von AS-Alternativen

Das Vorgehen der ökonomischen Bewertung von Informationssystemen, das sich in den vergangenen Jahren herausgebildet hat, ist eine reine Kostenbetrachtung. Ein hierzu häufig herangezogenes Verfahren ist die „*Total Cost of Ownership Analyse*“ (siehe Abschnitt 2.4.1). Zunächst werden in diesem Abschnitt Methoden zur Kostenbewertung vorgestellt. Darauf folgend wird die Notwendigkeit der Bewertung von Nutzen dargestellt und zwei methodische Ansätze erläutert. Im Anschluss an die Einführung von Methoden der Kosten- und Nutzenbewertung, werden Methoden zur multidimensionalen Bewertung vorgestellt und erläutert. Abschließend werden die Konsequenzen aufgezeigt und die Verfahren kritisch gewürdigt.

2.4.1 Kostenbewertung des Einsatzes von OSS

Die Schlagworte „*Kostenführerschaft*“ und „*Kostenreduktion*“ haben in den letzten Jahren unter anderem durch die breite Akzeptanz generischer Wettbewerbsstrategien (vgl. Porter, 1999, S. 34 ff.) an Bedeutung gewonnen. Häufig wird mit den vermeintlich geringen Beschaffungs- und Updatekosten von OSS argumentiert. Dies ist scheinbar das klarste und naheliegendste Argument für den Einsatz von OSS. Riklin (vgl. Riklin, 2006, S. 15) zeigt in einer 2006 durchgeführten Studie, dass 81% der Befragten „*Kostendruck*“ als größte informationstechnologische Herausforderung ansehen (siehe Abbildung 2.4). Der Vergleich von OSS und CSS war daher bisher häufig eine Gegenüberstellung der Kosten.

Das Fehlen der Lizenzkosten wird häufig als entscheidender Faktor für den Einsatz von OSS angesehen. Lizenzkosten stellen jedoch in einer TCO-Analyse (siehe Abschnitt 2.4.1) nur einen Teil der Gesamtkosten eines Anwendungssystems dar (vgl. Brügge et al., 2004, S. 116 ff.). Es ist daher

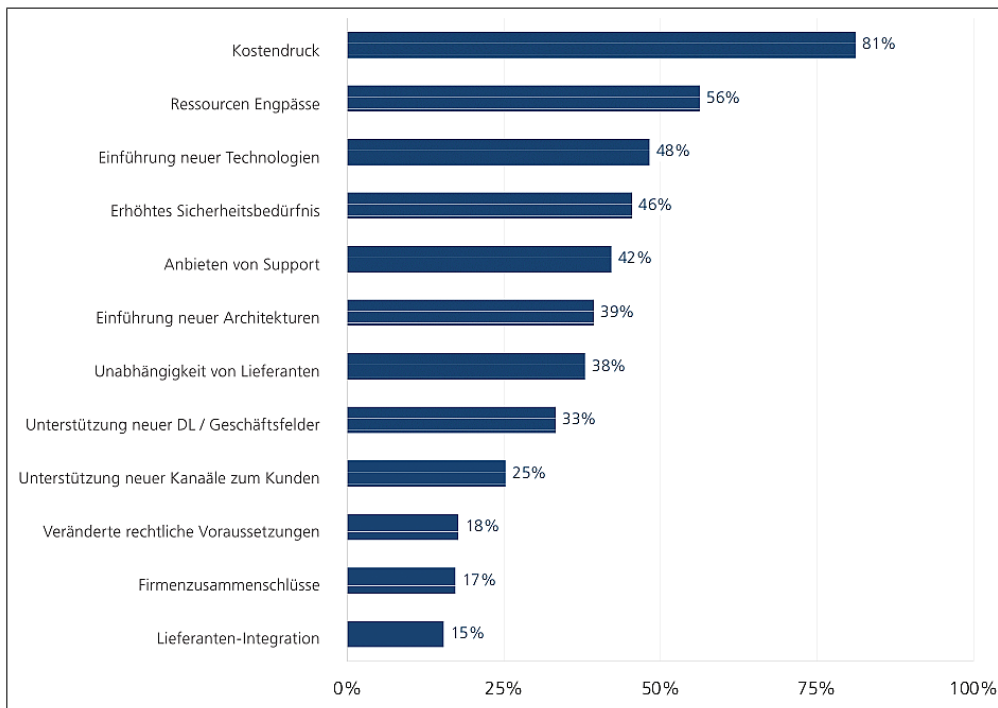


Abbildung 2.4: Herausforderungen in der IT nach Riklin (vgl. 2006, S. 15)

nicht sinnvoll, den Kostenvorteil, den OSS in diesem Bereich hat, als alleinige Entscheidungsgrundlage heranzuziehen. Selbst wenn die – gegenüber CSS ebenfalls häufig geringeren – Updatekosten von OSS bei einer Analyse Ansatz finden, genügt die reine Kostenbetrachtung nicht der Forderung nach einer umfassenden Bewertung.

Total Cost of Ownership

Das von der Gartner Group entwickelte „*Total Cost of Ownership*“-Verfahren ist ein kostenbasiertes Verfahrensmodell zur Bewertung von IT-Projekten. Bei der Analyse werden die Gesamtkosten eines IT-Projekts erhoben. Dabei wird das Projekt in zwei Hauptkostenarten bewertet. Es wird von direkten (haushaltswirksamen) und indirekten (nicht-haushaltswirksamen) Kosten gesprochen. Wie in Abbildung 2.5 dargestellt, unterteilen sich die direkten

Kosten dabei weiter in die Primärkostenarten Hard-/Software, Operations, Verwaltung und die indirekten Kosten in die Primärkostenarten End User Operations, Downtime (vgl. Hall, 2003). Der Lebenszyklus für die TCO-Analyse ist dabei nach Stahlknecht und Hasenkamp (vgl. Stahlknecht und Hasenkamp, 2005, S. 473) die gesamte Entwicklungs- und Nutzungsdauer des betrachteten Systems.

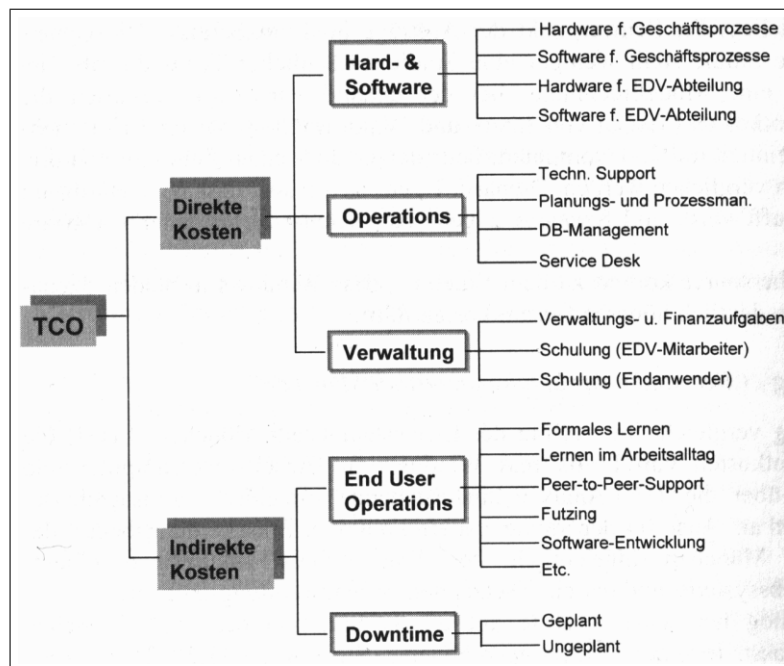


Abbildung 2.5: Total Cost of Ownership nach Brügge et al. (vgl. 2004, S. 117)

Zur Erhebung der Primärkostenart „Hardware/Software“ sind nach Bensberg et al. (vgl. Bensberg et al., 2007) sämtliche Hardware-/Softwarekosten der Bereitstellung des betrachteten AS für Anwenderprozesse und die Folgekosten der Leistungserstellung (Kosten für Software/Hardware der IT-Abteilung) zu erfassen. Eine weitere Systematisierung der Hardwarekosten in Kernprodukt, Ersatzteile, Upgrades, Betriebsstoffe und Software und der Software nach dem Anwenderbezug (direkte Unterstützung von Geschäftsprozessen, Leistungserstellung durch IT-Abteilung) wird im TCO-Modell nach Wild und Herges (vgl. Wild und Herges, 2000) vorgeschlagen.

Daraus folgt nach Bensberg et al. (vgl. Bensberg et al., 2007) die Bewertung periodenspezifischer Kostenanteile für Hard-/Software mit deren Abschreibungswerten. Unter operativen Kosten sind z. B. Aufwände für technischen Support, Planungs- und Prozessmanagement, Datenbankmanagement und Benutzersupport zu verstehen (siehe Abbildung 2.5). Unter Verwaltungsaufwand werden Kosten für organisatorisch-administrative Aufgaben der IT-Abteilung (Management IT-Budgets, Mitarbeiter) sowie Schulungskosten (Anwender, IT-Abteilung) zusammengefasst (vgl. Bensberg et al., 2007).

Als „indirekte“ Kosten werden zum einen Opportunitätskosten bezeichnet, die entweder durch nicht direkt dem Leistungserstellungsprozess zuordenbare Tätigkeiten der Anwender (Lernen, Support untereinander, private Nutzung, Entwicklung eigener Dokumente und Makros) oder Systemausfälle entstehen, wie Abbildung 2.5 ebenfalls zeigt.

Kosten werden in einer TCO-Analyse periodenbezogen erhoben und aufsummiert. Eine Diskontierung (z. B. Kapitalwertmethode) findet nicht statt. Die Durchführung einer TCO-Analyse kann mit einer herkömmlichen Tabellenkalkulation (z. B. Excel, Calc) oder mit Hilfe des vom schweizerischen Informatikstrategieorgan Bund (ISB) veröffentlichten TCO-Tool¹⁵ erfolgen.

Rechenmodell der EU - IDA

IDABC (Interoperable Delivery of European eGovernment Services to public Administrations, Businesses and Citizens)¹⁶ ist ein Projekt der Europäischen Kommission für die Verbesserung der Interoperabilität des Datenaustauschs und der Bereitstellungen von Informationen und Diensten unter öffentlichen Einrichtungen und gegenüber den Bürgern. Bereits im Rahmen des Projektvorgängers „Interchange of Data and Administrations“ (IDA) wurde mit den

¹⁵ <http://www.tcotool.org/>, zuletzt besucht am 5.8.2007

¹⁶ <http://ec.europa.eu/idabc/en/home>, besucht am 4.7.2007

„*Migration Guidelines*“ (vgl. Hnizdur, 2003) ein Rechenmodell zur Vorteilhaftigkeitsbewertung von OSS-Migrationen veröffentlicht.

Dieses Rechenmodell ist stark auf den Migrationscharakter eines OSS-Projektes ausgerichtet. Es erfasst dabei getrennt die Kosten einer proprietären Lösung und einer OSS-basierten Lösung sowie die durch das Migrationsvorhaben selbst entstehenden Kosten. Ziel des Modells ist es, einen Überblick über wesentliche Kostentreiber eines Migrationsprojektes zu schaffen. Dabei werden die CSS-Alternative und die OSS-Alternative sowie die jeweiligen Kosten während der nächsten fünf Nutzungsjahre betrachtet. Einmalige Kostenpositionen einer Migration sind in diesem Kostenmodell explizit abgegrenzt (vgl. Vriendt, 2003, S. 1 ff.). Für die proprietäre Lösung sieht das Kostenmodell ein Softwareupgrade während des Betrachtungszeitraumes vor. Die Kosten werden in den drei Bereichen getrennt nach „Hardware“, „Software“ und „Personal“ summiert und in einem Überblick dargestellt.

Das Rechenmodell ist in Tabellenform veröffentlicht und kann je nach konkretem Bedarf angepasst werden¹⁷.

2.4.2 Nutzenbewertung des Einsatzes von OSS

Bei den bisher betrachteten Analyseverfahren handelte es sich ausschließlich um Verfahren der Kostenbewertung. Die Einführung neuer Informationstechnologie wirft aber zwangsläufig die Frage nach den Wirkungsfeldern und dem Beitrag dieser Technologie zum Unternehmenserfolg auf. Fünf wesentliche Wirkungsarten (Kostenreduktion, Produktivitätssteigerung, Qualitätsverbesserung, Flexibilitätssteigerung und Verbesserung der Wettbewerbsposition) der Informationstechnologie sind dabei empirisch belegbar (vgl. Schumann, 2001a, S. 333). Die Schwierigkeit, den „Nutzen“ von IT-Investition zu erkennen und zu messen, stellt ein erhebliches Problem dar,

¹⁷ <http://ec.europa.eu/idabc/en/document/2623#migration>, besucht am 26.06.2007

wie Brynjolfsson (vgl. Brynjolfsson, 1993) bei der Untersuchung des „*Productivity Paradox of Information Technology*“ zeigt. Bestandteil einer fundierten Entscheidungsvorbereitung müssen deshalb neben direkt oder indirekt monetär messbaren Nutzen auch alle qualitativen und strategischen Nutzen sein. Da monetär messbare Nutzen innerhalb der Kostenbetrachtung in Form negativer Kosten bereits Ansatz finden können, werden im Folgenden Methoden zur qualitativen Bewertung von Nutzen erläutert. Hinsichtlich ihrer Bewertbarkeit ergibt sich nach Nagel (vgl. Nagel, 1990, S. 24 ff. , S.72) die Unterteilung des Nutzenbegriffs in drei Nutzenkategorien. Die direkten Nutzen sind zum Beispiel Kosteneinsparungen und werden durch Ansatz negativer Kosten erfasst. Indirekte Nutzen sind zukünftige Kosteneinsparungen oder Produktivitätsvorteile und werden, sofern abschätzbar, ebenfalls als abgezinste negative Kosten zukünftiger Perioden berücksichtigt. Gegenstand der Nutzenbewertung mit den im Folgenden vorgestellten Methoden sind daher nur die sogenannten schwer fassbaren, nicht quantifizierbaren qualitativen und strategischen Nutzen.

Nutzwertanalyse

Die Nutzwertanalyse ist ein Verfahren zur Wahl von Alternativen bei nicht monetär quantifizierbarem Nutzen. Sie besteht dabei aus fünf aufeinander aufbauenden Schritten.

1. **Zielkriterienbestimmung** Zielkriterien festlegen
2. **Zielkriteriengewichtung** Gewicht einzelner Zielkriterien festlegen
3. **Teilnutzenbestimmung** Beitrag einer Alternative zur Erfüllung eines Kriteriums bestimmen
4. **Nutzwertermittlung** Aufsummieren der Teilnutzen je Alternative
5. **Beurteilung der Vorteilhaftigkeit** Vergleich des Nutzwertes mit dem der Alternativen oder Vorgabewerte

Zunächst müssen die Zielkriterien festgelegt werden. Dabei ist darauf zu achten, dass Mehrfachnennungen vermieden werden und Zielkriterien weitestgehend voneinander unabhängig sind. Um diese zentralen Prämissen der Nutzwertanalyse zu gewährleisten, werden Einzelkriterien zu Kriteriengruppen zusammengefasst. Es empfiehlt sich eine Zielkriteriendefinition nach Stahlknecht und Nordhauss (vgl. Stahlknecht und Nordhauss, 1981, S. 109) wie folgt:

Mindestkriterien KO-Kriterien, jede Alternative muss diese erfüllen

Wunschkriterien wünschenswerte Kriterien, die nicht zwingend erfüllt sein müssen, aber relativ hohes Gewicht haben

Kannkriterien Kriterien deren Erfüllung relativ niedriges Gewicht hat

Im Anschluss werden Zielgewichte zunächst für Kriteriengruppen und danach für Einzelkriterien festgelegt. Es empfiehlt sich dabei zum Zwecke der einfacheren Berechnung eine Normierung der Kriteriengewichte vorzunehmen. (vgl. Biethahn et al., 1994, S. 348 ff.)

| Punkteskala | |
|-------------|----------------|
| 0 | nicht |
| 1 | äußerst gering |
| 2 | kaum |
| 3 | befriedigend |
| 4 | gut |
| 5 | sehr gut |

Tabelle 2.2: Punkteskala (vgl. Biethahn et al., 1994, S. 351)

Danach ist eine Verhältnisskala oder Intervallskala (vgl. Schwarze, 2001, S. 34) zur Bewertung festzulegen. Mit dieser Skala (siehe Tabelle 2.2) erfolgt darauf eine Teilnutzenbestimmung mit $N_i = G_i * E_i$ und die Nutzwertbestimmung mit $N_{ges} = \sum_{i=1}^n N_i$ pro Alternative. Die vorteilhafteste Alternative wird gemäß Maximalwert-Prinzip ausgewählt (vgl. Krems, 2004).

AHP

Der von Saaty (vgl. Saaty, 1980) entwickelte „*Analytical Hierarchy Process*“ (AHP) ist eine vor allem im angelsächsischen, skandinavischen und asiatischen Raum verbreitete Methode zur Bewertung von Alternativen bei multi-kriteriellen Entscheidungen. Methodisch handelt es sich um eine Erweiterung der Nutzwertanalyse durch paarweisen Vergleich von Kriterien. Es können sowohl qualitative als auch quantitative Kriterien bewertet werden. Der AHP besteht aus vier aufeinanderfolgenden Schritten.

1. **Problemdefinon** Defintion des Bewertungsproblems
2. **Kriterienhierarchie** Definition eines Oberziels und Ableiten einer Kriterienhierarchie
3. **Paarvergleich** Alle Kriterien einer Hierarchieebene werden mittels Paarvergleich in einer Matrix bewertet
4. **Eigenvektorisierung** Eigenvektorisierung der Matrix und Übertragung der Ergebnisse in eine obere Hierarchieebene

Durch das Ableiten einer Kriterienhierarchie im zweiten Schritt der Bewertung kann die bei Miller (vgl. Miller, 1956) nachgewiesene limitierte Kapazität des menschlichen Gehirns umgangen werden. Durch wiederholtes Quadrieren der Matrizen, Berechnen und Normieren der Reihensummen und Abweichungsvergleich zweier Iterationen mit Abbruchbedingung entsteht iterativ die Prioritätenordnung der Kriterien.

Da es sich im Falle einer inkonsistenten Paarvergleichsmatrix um eine Zufallsentscheidung handelt (vgl. Rohr, 2004, S. 46), muss für fundierte Entscheidungen ein weitgehend konsistentes Werturteil des Entscheidungsträgers vorhanden sein. Vollständige Konsistenz ist dabei im Idealfall nur bei 2×2 Matrizen, also beim Vergleich lediglich zweier Alternativen möglich. Es wird

daher ein Konsistenzindex auf Basis des Vergleichs zwischen dem Eigenwert der ermittelten Matrix und dem maximalen Eigenwert berechnet.

Auf eine Darstellung der mathematischen Grundlagen soll an dieser Stelle verzichtet werden, da sie den Rahmen der Arbeit sprengen würde. Eine detaillierte Darstellung findet sich zum Beispiel bei Rohr (vgl. Rohr, 2004, S. 39 ff.).

In der Gesamtwürdigung der Methode ist festzustellen, dass der AHP ein mathematisch zwar wesentlich aufwendigeres Verfahren ist, dafür aber ein genaueres Ergebnis liefert als die Nutzwertanalyse. Obgleich nicht ausgeschlossen ist, dass beide Verfahren dieselbe Alternativenreihenfolge ergeben, kann auf Grund der dargestellten Komplexitätsreduktion vom AHP eine „bessere“ Entscheidungsgrundlage erwartet werden. Kritisch ist anzumerken, dass sich bei ständiger Änderung der Umwelt, durch Zutritt oder Wegfall von Kriterien im dynamischen Projektumfeld die relative „Starrheit“ des Verfahrens negativ auswirkt und es durch erneuten Paarvergleich der Kriterien zu einer Invertierung der Alternativenreihenfolge kommen kann¹⁸ (vgl. Harker und Vargas, 1990, S. 270 ff.). Zumindest bei sorgfältig mit den strategischen Zielen abgestimmten Bewertungskriterien und mittelfristiger Beibehaltung der strategischen Ziele ist die Auswirkung jedoch begrenzt.

2.4.3 Wirtschaftlichkeitsbewertung

Anwendungssysteme stellen aus betriebswirtschaftlicher Sicht Realinvestitionsobjekte dar. Zur Rechtfertigung von Investitionsentscheidungen sind daher nicht nur die Kosten/Aufwände, sondern auch die Leistungen/Erträge eines Anwendungssystems zu bewerten. In diesem Kontext wird von der Beurteilung der Wirtschaftlichkeit gesprochen (vgl. Biethahn et al., 1994, S. 332 f.). Kosten und monetäre Nutzen eines Anwendungssystems lassen sich relativ einfach mit den in 2.4.1 vorgestellten Methoden bewerten. In 2.4.2

¹⁸ engl. Rank Reversal

wurden Methoden zur Bewertung nicht-monetär erfassbarer qualitativer und strategischer Nutzen vorgestellt.

Die im Folgenden vorgestellten Wirtschaftlichkeitsbewertungsverfahren dienen der vollständigen Entscheidungsvorbereitung von IT-Projekten. Sie werden als multidimensional bezeichnet, weil sie über die reine Kostenbewertung hinaus mindestens auch eine Bewertung monetärer Nutzen ermöglichen.

WiBe

Die „*Koordinierungs- und Beratungsstelle der Bundesregierung für Informationstechnik in der Bundesverwaltung*“ (KBSt) hat ebenfalls ein Verfahren zur Bewertung von OSS-Migrationsprojekten unter dem Namen „*Wirtschaftlichkeitsbetrachtung (WiBe)*“ veröffentlicht. Durch das Verfahren werden neben quantitativ messbaren Faktoren eines Projektes auch die qualitativen und strategischen Faktoren bewertet.

Das Verfahren läuft in drei Schritten ab:

1. Selektion relevanter Kriterien für die WiBe
2. Durchführung der Datenerhebung
3. Wirtschaftliche Beurteilung des Projekts

Der sogenannte generelle Kriterienkataloge umfasst 4 Kriterienbereiche. Diese werden in „Wirtschaftlichkeit im engeren Sinne“ (monetär quantifizierbare Faktoren) und „Wirtschaftlichkeit im weiteren Sinne“ (qualitativ-strategische Faktoren) unterteilt.

1. Wirtschaftlichkeit im monetären Sinne (WiBe KN)
2. Dringlichkeit des IT-Projekts (WiBe D)

3. Qualitativ-strategische Bedeutung des IT-Projekts (WiBe Q)
4. Externe Effekte des IT-Projekts (WiBe E)

Im Rahmen der Wirtschaftlichkeit im monetären Sinne lassen sich alle quantifizierbaren Kosten und Nutzen eines IT-Projekts erfassen. Dabei können sowohl Kosten als auch Nutzen entweder einmalig oder laufend während des Betrachtungszeitraumes anfallen. Zunächst ist zu diesem Zweck der generelle Kriterienkatalog (vgl. Röthig et al., 2004, S. 19 ff.) in den Bereichen „Entwicklungskosten/Entwicklungsnutzen“ und „Betriebskosten/Betriebsnutzen“ auf die Erfordernisse des speziellen IT-Projekts anzupassen. Im Folgenden wird der Kriterienkatalog für die Dringlichkeit des IT-Projekts (z. B. Instabilität des Altsystems, Auslaufender Produktsupport), qualitativ-strategische Faktoren (z. B. Herstellerunabhängigkeit, Standardisierung u. a.) und externe Faktoren (z. B. Benutzerfreundlichkeit aus Kundensicht, Mehrwerte für den Kunden) ebenfalls an das spezielle IT-Projekt angepasst.

Im Anschluss an das Anpassen des Kriterienkataloges (dieser kann im Grenzfall sogar ganz neu angelegt werden) erfolgt die Datenerhebung in den Einzelbereichen. Hinsichtlich ihrer Höhe unsichere, monetäre Werte können mit einem Risikozuschlag/Risikoabschlag bewertet werden (vgl. Röthig et al., 2004, S. 26). Im Rahmen des Modells wird dann von WiBe KN/R gesprochen. Analog zur monetären Datenerhebung werden im Anschluss die Daten in den Bereichen WiBe D, WiBe Q und WiBe E erhoben. Ein Leitfaden zur Erhebung ist bei Röthig et al. (vgl. Röthig et al., 2004, S. 30 ff.) mit Bezug auf den generellen Kriterienkatalog veröffentlicht.

Die Berechnung der Wirtschaftlichkeit des IT-Projekts erfolgt im Bereich WiBe KN bzw. WiBe KN/R bezugsgenau durch Verwendung der Kapitalwertmethode $C_0 = \sum_{t=0}^T N_t * q^{-t}$ (vgl. Biethahn et al., 1994, S. 340). Die qualitativ-strategischen Faktoren werden mit der Nutzwertanalyse (siehe Abschnitt 2.4.2) bewertet.

Das Verfahren lässt sich durch Anpassung der Kriterienkataloge auf sehr spezielle IT-Projekte zuschneiden (vgl. Leipelt, 2000, S. 1. ff). Es ist darüber hinaus in Abhängigkeit vom Umfang des Projektes während des gesamten Planungszeitraumes und im Betrieb als Planungs- bzw. Kontrollwerkzeug einsetzbar. Nach dem Fachkonzept erfolgt eine Bewertung mit der Erstellung des Grobkonzeptes (Version 1), mit Erstellung des Feinkonzeptes (Version 2), unmittelbar vor der Einführung (Version 3) und nach Inbetriebnahme (Version 4) (vgl. Röthig et al., 2004, S. 11 f.).



Abbildung 2.6: Portfolioanalyse

Um letztlich auf Basis der monetären und der qualitativ-strategischen Bewertung eine Beurteilung der Vorteilhaftigkeit der Alternativen treffen zu können, bietet sich die im Informationsmanagement häufig verwendete Portfoliotechnik (vgl. Schumann, 2001b, S. 251) an. Die Portfoliotechnik setzt sich nach Wilde (vgl. Wilde, 1992, S. 212 ff.) zunehmend als Standardverfahren zur Entwicklung und Visualisierung von Normstrategien durch. Die Normstrategien ergeben sich dabei nach Schumann (vgl. Schumann, 2001b, S. 251

f.) laut den auf der x-Achse und der y-Achse dargestellten Dimensionen in einer 2*2 oder 3*3 Matrix (siehe Abbildung 2.6). In der aktuell veröffentlichten Version (WiBe 4.0) des WiBe-Verfahrens wird zur Gegenüberstellung der Bewertungsergebnisse der „Wirtschaftlichkeit im engeren Sinne“ und „Wirtschaftlichkeit im weiteren Sinne“ die Portfolioanalyse verwendet¹⁹.

TEI

Total Economic Impact (TEI) ist ein von der Giga Group Inc. entwickeltes Verfahren zur Transparenzsteigerung von IT-Projekten. Es besteht aus einer vier Teilbereiche umfassenden Bewertung (siehe Abbildung 2.7).

1. **Benefits** Darstellung des Nutzenbeitrag des geplanten Projektes
2. **Costs** Sammlung aller im Projektzusammenhang anfallenden Kosten
3. **Flexibility** optionale Investitionen
4. **Risk** Risikobewertung aller Kosten, Nutzen und optionaler Investitionen

In der TEI-Analyse werden Kosten, Nutzen und Investitionsoptionen monetär bewertet. Auf Basis einer Risikoabschätzung wird der „Return on Investment“ für das gesamte IT-Projekt ermittelt. Das Verfahren hat aus diesem Grunde eine rein monetäre Ausprägung.

Die TEI-Analyse startet mit der Erhebung potentieller Nutzenfelder. Unter „*Benefits*“ sind die monetär messbaren Nutzenbeiträge eines Projekts zu verstehen. Für diese Nutzen wird durch TEI jedoch ein Nachweis über kritische Erfolgsfaktoren und damit eine erkennbare Verbindung zu einer übergreifenden Unternehmensstrategie gefordert. Nutzen können dabei nur

¹⁹ http://www.kbst.bund.de/cln_046/nn_837402/Content/Wirtschaft__u__Recht/Wirtschaftlichkeit/wirtschaftlichkeit__node.html__nnn=true#doc838072bodyText1, besucht am 26.06.2007

auf Geschäftsbereichsebene wirken. Jeder Nutzen muss von einem Experten innerhalb des Unternehmens hinsichtlich seines monetären Wertes und eines Eintrittsdatums bestimmt werden. Dabei werden Nutzen, die über einen längeren Zeitraum anfallen, zu dem Zeitpunkt angesetzt, zu dem ihre überwiegende Realisierung erreicht wurde. Nutzen, welche sich z. B. durch Änderung an den Geschäftsprozessen ergeben, müssen von Managern bewertet werden, die solche Änderungen auch beschließen können. Unter „Costs“ werden alle Kosten zusammengefasst, die zur Erschließung der Nutzen anfallen. Dazu gehören insbesondere Arbeitskosten, Dienstleisterkosten und Soft-/Hardwareinvestitionen, aber auch direkte Ersparnisse aus dem Einsatz neuer IT-Systeme (in Form negativer Kosten).

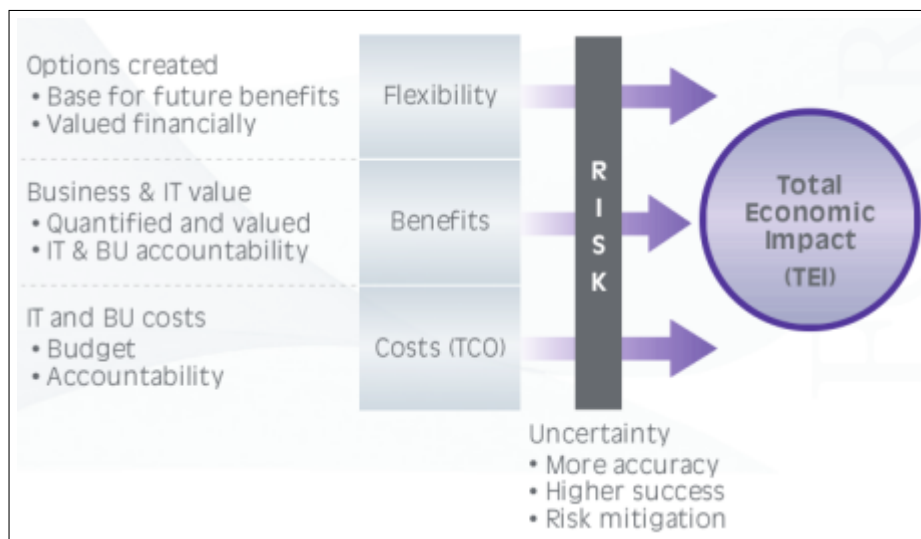


Abbildung 2.7: Total Economic Impact (vgl. Colony, 2007)

In der TEI-Analyse wird ein IT-Projekt in die „benefits phase“ und die „option phases“ unterteilt. Innerhalb der „benefits phase“ werden die Hauptnutzen realisiert. Die darauf folgenden „option phases“ erlauben dem Management monetär bewertete Investitionsoptionen in die Bewertung aufzunehmen, deren Wert mit der „Black-Scholes option pricing equitation“ nach Black und Scholes (vgl. Black und Scholes, 1973) unter Berücksichtigung des branchenüblichen Kapitalmarktzinses berechnet wird.

Um inhärente Projektrisiken in der Bewertung zu berücksichtigen, werden sowohl die erfassten Kosten als auch die Nutzen und Optionen durch eine Risikoanalyse mittels dreistufiger (worst case, normal case, best case) Einschätzung und Bewertung durch eine Wahrscheinlichkeitsdichtefunktion dargestellt (vgl. Rymer und Cormier, 2003, S. 20 ff.).

2.4.4 Konsequenzen ökonomischer Bewertungsverfahren

Das in Abschnitt 2.4.1 vorgestellte TCO-Verfahren stellt das Standardverfahren zur Kostenbewertung von IT-Investitionen dar. Kritisiert wird es vor allem wegen des zu kurzen Betrachtungszeitraumes (3 Jahre), der Fundierung auf empirischen Daten des Beratungsunternehmens Gartner (vgl. Kofler, 2004, S. 8) und der fehlenden Diskontierung zeitversetzt anfallender Kosten. Das Rechenmodell der EU-IDA (siehe Abschnitt 2.4.1) ist ebenfalls ein kostenorientierter Ansatz, der den besonderen Migrationscharakter vieler OSS-Projekte berücksichtigt. Insbesondere wird der Betrachtungszeitraum auf 5 Jahre erweitert, Kostenpositionen zeitbezogen diskontiert (Kapitalwert) und ein möglicher Softwareupgradezyklus berücksichtigt, daher wurde dieses Verfahren alternativ vorgestellt.

Sowohl der Total Cost of Ownership Ansatz als auch das Rechenmodell der EU-IDA ermöglichen zwar eine vollständige Kostendarstellung (vgl. Brügge et al., 2004, S. 117), im Rahmen der vollständigen Wirtschaftlichkeitsbetrachtung müssen jedoch auch monetäre Nutzen Berücksichtigung finden. Wie aus Abschnitt 2.4.1 und Abschnitt 2.4.2 hervorgeht, ist allein die Erhebung von Kosten oder Nutzen eines IT-Projekts nicht ausreichend.

Das TEI-Verfahren (siehe Abschnitt 2.4.3) entstand als Erweiterung der in den USA üblichen Kostenbewertung durch das TCO-Verfahren (siehe Abschnitt 2.4.1). Es basiert auf der vollständig monetären Bewertung eines IT-Projekts und erfasst neben den inhärenten Risiken auch spätere Inves-

titionsoptionen während der Betriebsphase des betrachteten AS. Das Bewertungsverfahren setzt eine vollständig monetäre Bewertbarkeit der Nutzen voraus. Übergreifende strategische Nutzen werden in der TEI-Analyse allerdings nicht angesetzt.

Die Güte der Bewertung von Investitionsoptionen nach der „*Black-Scholes option pricing equitation*“ wurde durch die empirische Ermittlung systematischer Fehlbewertung im Bereich des Aktienmarktes (vgl. Pape und Merk, 2003, S. 4) häufig angezweifelt. Dennoch findet das Verfahren gerade in den USA gerne Anwendung (vgl. Day und Erickson, 2001; Rymer und Cormier, 2003).

Wie in Abschnitt 2.4.2 dargestellt wurde, müssen bei einer vollständigen Nutzenbewertung insbesondere auch Methoden zur Bewertung nicht monetär erfassbarer Nutzen verwendet werden. Hierzu wurden sowohl das in Europa gängige Verfahren der Nutzwertanalyse (siehe Abschnitt 2.4.2) als auch dessen methodische Erweiterung AHP (siehe Abschnitt 2.4.2) vorgestellt. Beide Verfahren ermöglichen die Bewertung qualitativer und strategischer Nutzen. Die Nutzwertanalyse verzichtet zu Gunsten mathematischer Einfachheit auf die inhärente Prüfung eines konsistenten Werturteils.

Die nach Schumann (vgl. Schumann, 2001c, S. 504) „*umfassendste Information als Entscheidungsgrundlage*“ kann nur durch die Kombination der Erfassung von Kosten und Nutzen und deren Bewertung mit einem multidimensionalen Bewertungsverfahren mit einer sogenannten Wirtschaftlichkeitsbewertung erfolgen. Insbesondere Ziele bzw. Nutzen, welche sich aus den strategischen Faktoren des Einsatzes von OSS (siehe Abschnitt 2.2.4) und den im Abschnitt 2.3.5 dargestellten „Konsequenzen technischer Grundlagen von Anwendungssystemen“ ergeben, lassen sich dabei häufig nicht monetär bewerten.

Das in der Praxis der öffentlichen Verwaltung bewährte WiBe-Verfahren (siehe Abschnitt 2.4.3) besteht aus der Kombination von Gesamtkosten- und Gesamtnutzenerhebung. Nicht monetär erfassbare Nutzen werden mit der Nutz-

wertanalyse bewertet und in einer Portfoliodarstellung mit den monetären Kosten/Nutzen verglichen.

Der mit dem WiBe-Verfahren (siehe Abschnitt 2.4.3) veröffentlichte generelle Kriterienkatalog erweitert die reine Kostendarstellung (WiBe KN / WiBe KN/R) um die Darstellung monetär quantifizierbarer Nutzen. Der Kriterienkatalog ist dabei nicht statisch, sondern kann flexibel an das jeweilige Projekt angepasst werden. Dabei sind Kosten und Nutzen nicht nur über einen kurzen Zeitraum zu bewerten²⁰. Ein zu kurzer Betrachtungszeitraum würde u. U. später eintretende Nutzeneffekte nicht berücksichtigen und die Entscheidung damit verfälschen. Die beiden in Abschnitt 2.4.1 vorgestellten Kostenmodelle unterstellen Sicherheit in Verbindung mit der Kosten und Nutzenbewertung. Besonders externe Kosten-/Nutzeneffekte (Dienstleister, Lizenzkosten von CSS u. a.) oder Kosten-/Nutzeffekte, welche auf Grund von Mangel an empirischen Werten geschätzt werden, müssen jedoch mit einem Risikozuschlag versehen werden können. Das WiBe-Verfahren ermöglicht eine Berücksichtigung solcher Risiken und stellt die Kosten-/Nutzeneffekte dabei mit dem Kapitalwertverfahren diskontiert dar.

Das WiBe-Verfahren deckt damit als einziges die geforderte vollständige Kosten- und Nutzenbetrachtung ab. Es ist darüber hinaus sehr flexibel an konkrete Projektausprägungen anpassbar. Da es öffentlich zugänglich dokumentiert ist (vgl. Röthig et al., 2004) und eine Softwareimplementierung (WiBe 4.0) ebenfalls öffentlich zugänglich existiert²¹, ist es transparent und aktuell.

Dieser Vollständigkeit und der „offenen“ Ausrichtung wegen, erhält das WiBe-Verfahren in dieser Arbeit den Vorzug gegenüber dem TEI-Ansatz.

²⁰ je nach Umfang des Projekts können Migrationskosten sehr hoch sein

²¹ wird vom Bundesministerium des Inneren (BMI) ständig aktualisiert und dokumentiert

Kapitel 3

Entscheidungsmodell

Bei einer detaillierten Analyse dokumentierter Migrationsprojekte zum Einsatz von Open Source Software, vorwiegend aus der öffentlichen Verwaltung, aber auch der freien Wirtschaft, ist eine in weiten Teilen gleiche Vorgehensstruktur festzustellen, die im Wesentlichen die Elemente *Aufnahme der IST-Situation*, *Technische Machbarkeit*, *Monetäre Bewertung*, sowie *Qualitativ-strategische Bewertung* umfasst. Dies entspricht dem beim Aufbau komplexer technischer Systeme üblichen „Phasenmodell“ (vgl. Ferstl, 2001, S. 337) und ist deshalb auch als Grundlage des hier vorgestellten Entscheidungsmodells genutzt. Dieser Grundstruktur wird eine Projektvorstudienphase und eine Entscheidungsphase hinzugefügt. Auf Basis einer detaillierten Situationsbestimmung werden zunächst die Alternativen anhand einer technischen Machbarkeitsanalyse ermittelt.

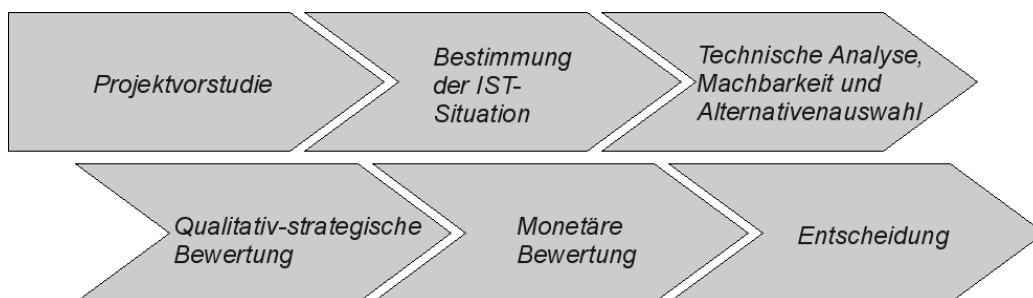
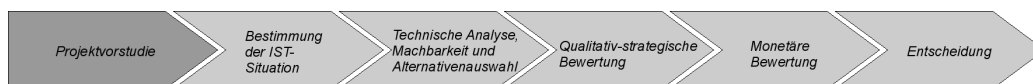


Abbildung 3.1: Entscheidungsmodell

Anschließend werden die ermittelten Alternativen monetär bzw. hinsichtlich ihrer Auswirkung auf die Erreichbarkeit qualitativer und strategischer Ziele bewertet. Obwohl das beschriebene Vorgehen die Entscheidung schrittweise und aufeinander basierend begründet, fehlt in vielen dokumentierten Fallbeispielen¹ eine abwägende, gegenüberstellende Gewichtung der monetären, qualitativen und strategischen Bewertung und/oder eine darauf basierende Entscheidungsregel. Ebenso fehlt oft eine detaillierte Projektvorstudie zur Abgrenzung des Projektumfangs und Erhöhung der Projekterfolgswahrscheinlichkeit. Das in dieser Arbeit entwickelte Entscheidungsmodell enthält diese Elemente. Durch die Unterteilung in sechs Entscheidungsfindungsphasen und die Vermeidung unnötiger, nicht zielführender Analysen mit der Projektvorstudie, leistet bereits der Aufbau des Modells einen Beitrag zum begleitenden Projektmanagement (siehe Abschnitt 2.1.2) und der damit verbundenen Handhabung von Projektrisiken.

3.1 Projektvorstudie



Entscheidungsmodell - Phase 1

Das Ziel der Projektvorstudie ist eine möglichst genaue Projektbeschreibung des zu prüfenden IT-Projekts. Das betroffene Anwendungssystem soll hinsichtlich beteiligter Hard- und Software genau beschrieben werden. Zur Verfügung stehende Projektressourcen müssen, soweit möglich, genau ermittelt und inhärente Projektrisikofaktoren erstmalig benannt werden. Darüber hinaus werden qualitative und strategische Ziele für das Projekt benannt. Dieses Vorgehen ergibt einen Überblick über die Reichweite der zu treffenden Entscheidung, sowie einen ersten Gesamtüberblick über das Projekt.

¹ (vgl. Werner, 2007; Meyer und Brunke, 2005; Kunath, 2005; Hoegner et al., 2003; Lutz et al., 2004, S. 239 ff., S. 51 ff., S.25 ff., S. 6 ff., S. 8 ff.)

Durch die Beschreibung läßt sich eine erste Einschätzung der Komplexität des Projekts treffen und der Projektumfang eingrenzen. Eingehende, beziehungsweise ausgehende, zwingende Abhängigkeiten (Software bedingt andere Software, Hardware bedingt spezielle Software) können bereits in dieser Phase zu einer Anpassung des Projektumfangs oder gar der Ablehnung führen. Später erkannt, würden sie hohe Zusatzkosten bedeuten.

In dieser frühen Phase lassen sich also bereits Faktoren ermitteln, die zu einer klaren Entscheidung für oder gegen das Migrations-, / Substitutionsvorhaben sprechen. Darüber hinaus ermöglicht die bereits in dieser Entscheidungsphase durchgeführte erste Risikobestimmung und -bewertung, die besonderen Risikofaktoren in IT-Projekten zu erkennen und Maßnahmen zur Reduzierung/Eliminierung zu ergreifen. Eine hohe inhaltliche Durchführungsqualität bereits im ersten Projektschritt der Entscheidungsvorbereitung ist im Sinne eines zielführenden Projektmanagements und im Sinne effizienter Nutzung der Ressourcen zu fordern. Eine bessere Risikoeinschätzung und -vermeidung und die Verbesserung der Entscheidungsqualität sind die wichtigsten Ergebnisse der Projektvorstudie.

Ein risikominimierender Beitrag wird erreicht, weil der Entscheidungsvorbereitungstab bei einer späteren Durchführung i. d. R. beteiligt ist. Er kennt damit vor dem eigentlichen Projektbeginn bereits Umfang, Risikofaktoren und Ziele und kann diese gut einschätzen.

3.1.1 Ermittlung von Projektressourcen

Zur erfolgreichen Durchführung eines Projektes ist ein effektives Management der zur Verfügung stehenden Projektressourcen notwendig. Häufig kommen in IT-Projekten nicht nur eigene Ressourcen, also eigene Mitarbeiter, Betriebsmittel und Kapital zum Einsatz, sondern insbesondere bei Know-how intensiven Spezialprojekten auch externe Dienstleister, die projektbezogenes Entwicklungs- bzw. Durchführungs-Know-how mitbringen.

Neben der Einbindung der Nutzer, die während eines IT-Projektes zu jeder Zeit gewährleistet sein sollte, ist die Unterstützung durch die Unternehmensleitung ein zentraler Erfolgsfaktor, wie die Chaos Studie der Standish Group (vgl. Gaulke, 2004, S. 43) gezeigt hat. Nur wenn das Erreichen strategischer Unternehmensziele durch die IT-Strategie und deren konkrete Maßnahmen (IT-Projekte) unterstützt wird, ist eine strategische Übereinstimmung gewährleistet (vgl. Gaulke, 2004, S. 14).

Inhaltliche Überzeugungsarbeit durch Projektleitung und strategische Übereinstimmung stellt die nachhaltige Unterstützung durch die Unternehmensleitung sicher. Neben der Verfügbarkeit personeller Ressourcen ist das im Unternehmen vorhandene und verfügbare Know-how sorgfältig zu ermitteln. Da bei Migrationsprojekten meist nicht nur Softwareentwicklung betrieben werden muss, sondern in der Regel auch mehr oder weniger große, häufig unternehmenskritische Datenbestände bewegt werden müssen, erfordert das Management solcher Projekte bereits bei der Entscheidungsfindung ein hohes Maß an projektspezifischem Know-how.

Das notwendige Know-how kann entweder durch externe Dienstleister oder durch Einstellung von geeignetem Personal beschafft werden. Da bereits in der Projektvorstudie strategische Überlegungen über die zukünftige Personalstruktur entscheidend in die spätere ökonomische Bewertung eingehen, ist auch dort mit hoher Sorgfalt zu ermitteln.

Dabei wird deutlich, dass der Aufbau bzw. die Beschaffung von geeignetem Know-how für erste Migrationsprojekte zu OSS zunächst erhebliche Kosten verursacht. Als äußerst zweckmäßig hat es sich erwiesen, externe Dienstleister einzubinden, welche das geplante Migrationsvorhaben durch ihr Know-how unterstützen und so den Transfer des Know-hows während der Durchführungsphase und auch in der sich anschließenden Betreuungsphase sicherstellen, wie die Fallbeispiele der Norddeutschen Affinerie (vgl. Meyer und Brunke, 2005, S. 52 ff.), der Stasi-Unterlagenbehörde BStU (vgl.

Kunath, 2005, S. 26 ff.) und der Landeshauptstadt München (vgl. Hoegner et al., 2003) zeigen.

Neben dem Gewinnen der Unterstützung durch die Unternehmensleitung ist im Rahmen der entscheidungsvorbereitenden Projektvorstudie zu prüfen, welches Budget im Haushaltsplan des Unternehmens generell für derartige IT-Projekte zur Verfügung steht bzw. zur Verfügung gestellt werden kann. Zwar hat eine gute Entscheidungsvorbereitung in der Regel das Potential, für eine erweiterte Mittelzuteilung zu sorgen, allerdings sollte schon zu Beginn der Entscheidungsvorbereitung klar sein, in welcher Größenordnung Mittel bereitgestellt werden können. Auch die sinnvollsten IT-Projekte können nicht durchgeführt werden, wenn die Mittel zur Anfangsinvestition, die üblicherweise bei solchen Projekten anfallen, nicht verfügbar sind.

3.1.2 Abgrenzung des Untersuchungsobjekts

Zumindest bei der öffentlichen Verwaltung sorgte die Ankündigung des auslaufenden Produktsupports für Windows oder Novell Software für erste Migrationsprojekte hin zu OSS. Da diese Projekte ausnahmslos Anfang der 2000er Jahre durchgeführt wurden, muss das zum aktuellen Zeitpunkt nicht mehr der Hauptgrund für Migrationvorhaben sein. Immer häufiger wird OSS-Software zum Beispiel deswegen eingesetzt, weil dadurch Qualitätsverbesserungen (siehe Abschnitt 2.3.4, 2.1.1) oder eine allgemein strategische Ausrichtung (siehe Abschnitt 2.2) der Informationsinfrastruktur erreicht werden kann. Die Gründe sind für die Prüfung einer Migration bei der Gesamtbetrachtung gleichwohl bedeutsam. Im Falle der Aufkündigung des sicherheits- und stabilitätskritischen Supports (Patches, Sicherheitsupdates) muss in jedem Fall migriert werden, während das bei rein strategischen Gründen nicht zwingend der Fall ist.

Anwendungssysteme (siehe Abschnitt 2.1.1) können je nach betrieblichem Anwendungsgebiet sehr komplexe technische Systeme sein. Oft können sie

daher im Rahmen eines vernünftigen Risikomanagements kaum als Ganzes betrachtet werden. Eine Migration des gesamten Anwendungssystems auf OSS wird darüber hinaus weder im Hinblick auf die qualitativ-strategische noch auf die ökonomische Bewertung sinnvoll oder wegen fehlender OSS-Komponenten gar nicht möglich sein.

In Anbetracht der Erkenntnis, dass neben der Projektdauer und der Teamgröße die Projektgröße ein entscheidender Erfolgsfaktor für IT-Projekte ist (vgl. Gaulke, 2004, S. 79 ff.), muss ein Bestandteil der Projektvorstudie sein, den Umfang des Untersuchungsobjekts entsprechend einzugrenzen. Dies schließt bereits zu Beginn der Entscheidungsvorbereitungsphase inhärente Projektrisiken von zu großen Projekten, wie zum Beispiel Verlust der Kontrolle über die Projektorganisation, den Projektfortschritt und die Projektergebnisse, aus (vgl. Gaulke, 2004, S. 79).

Neben der Ermittlung der personellen und finanziellen Projektressourcen (siehe Abschnitt 3.1.1) müssen bei Migrations- und Substitutionsprojekten die technischen und organisatorischen Abhängigkeiten geprüft werden. Insbesondere ist das Untersuchungsobjekt auf zwingende hardware- und softwareseitige Voraussetzungen bzw. nachgelagerte Abhängigkeiten zu prüfen, da diese sofort eine Anpassung bzw. Änderung des Projektumfanges oder gar eine Ablehnung zur Folge haben können.

Sind die Abhängigkeiten nicht zwingend, dient das Erfassen der Schnittstellen zu anderen Anwendungssystemen bzw. die Erfassung aktuell beteiligter Hardware der Komplexitätseinschätzung des Untersuchungsobjekts. Bei der Projektvorstudie muss ebenfalls untersucht werden, wie viele eigenständige Unternehmensteile und/oder Unternehmensstandorte durch das geplante Projekt betroffen sind. Bereits bei der Projektvorstudie sind auch externe Abhängigkeiten zu untersuchen. Insbesondere rechtliche Abhängigkeiten – dazu können zum Beispiel auch schwebende oder unklare Gesetzgebungsverfahren gehören – oder Abhängigkeiten zu anderen Projekten sind zu ermitteln.

3.1.3 Risikobestimmung des Projektes

Grundsätzlich stehen verschiedene unterstützende Methoden zur Risikobestimmung in IT-Projekten zur Verfügung. Neben der Befragung ausgewählter Experten und Analysen projektbezogener Dokumente können Kreativitätstechniken² bei der Durchführung von Workshops zur Bestimmung des Risikos eingesetzt werden (vgl. Gaulke, 2004, S. 66). Die Expertenmeinung des verantwortlichen Entscheidungsvorbereitungsstabes genügt in der Projektvorstudie, da dort lediglich eine erste Bestimmung inhärenter Projektrisiken vorgenommen werden soll, die ein umfassendes Risikomanagement in der Vorbereitung und während der Durchführung des Projekts keinesfalls ersetzen kann. Da in erster Linie ein unnötiges Durchlaufen der Entscheidungsphasen zwei bis sechs ausgeschlossen werden soll, ist dieses Vorgehen adäquat zum entstehenden Nutzen.

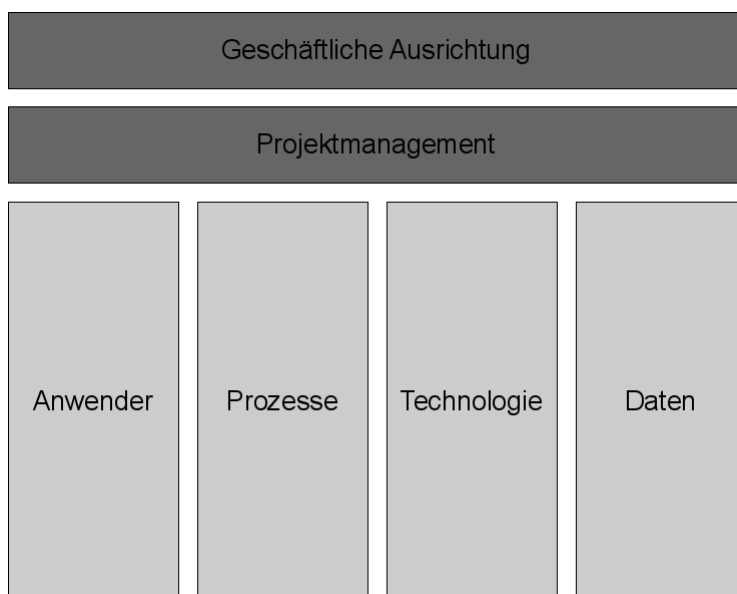


Abbildung 3.2: Risikobereiche – eigene Darstellung nach Gaulke (vgl. Gaulke, 2004, S. 60)

² z.B. Brainwriting, Brain Storming, Mindmapping

Die bei Gaulke (vgl. Gaulke, 2004, S. 58 ff.) beschriebene Unterteilung in sechs Projektrisikobereiche (siehe Abbildung 3.2) dient in der Projektvorstudie als Basis des Risikoaudits. Die ebenfalls beschriebenen Projektrisikochecklisten bilden, angepasst um Risikofaktoren aus den Schwerpunkten *Ermittlung von Projektressourcen* und *Abgrenzung des Untersuchungsobjekts*, die methodische Grundlage zur Erhebung einer Projektrisikokennzahl (siehe Anhang B.1).

Die dort vorgestellte Checkliste ist im Hinblick auf die Gesamtheit denkbarer Projektausprägungen nicht vollständig, kann aber entsprechend um jeweilige Projektspezifika betreffende Risikofaktoren erweitert werden. Bewertet werden die einzelnen Faktoren durch den Entscheidungsvorbereitungsstab. Dieses vorgeschlagene Vorgehen widerspricht zwar der grundsätzlichen Auffassung, IT-Risikomanagement sollte von Projektunbeteiligten durchgeführt werden (Projektbeteiligte neigen zu sehr zu optimistischen Risikoeinschätzungen (vgl. Gaulke, 2004, S. 61)), wird aber bei der ersten Risikoeinschätzung in Phase 1 bewusst in Kauf genommen. Dies ergibt zur unmittelbar vor Beginn des Projektes erneut, dann möglichst extern, durchzuführenden Risikobewertung einen zusätzlichen Indikator.

Die Bewertung selbst erfolgt mit einer Skala von „eins“ bis „fünf“, wobei „eins“ den geringsten und „fünf“ den höchsten Risikowert darstellen. Risikowerte, die mit vernünftigem Aufwand nicht zu erheben oder einzuschätzen sind, bleiben unbewertet. Der Risikowert eines Risikobereiches wird als arithmetisches Mittel aus den jeweils bewerteten Risikofaktoren innerhalb des Projektrisikobereichs ermittelt. Der Gesamtrisikowert wird im Anschluss als arithmetisches Mittel aus Mitteln der Projektrisikobereiche berechnet und ist ein zentrales Ergebnis des Projektvorstudienberichts.

$$\text{Arithmetisches Mittel} = \bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$$

3.1.4 Bestimmung qualitativer und strategischer Ziele

Neben den monetär bewertbaren Merkmalen und Eigenschaften gibt es eine Reihe qualitativer und strategischer Ziele, deren Erfüllung nur mit einer entsprechenden Analyse und nicht rein monetär oder technisch bewertet werden können. Hoegner et al. (vgl. Hoegner et al., 2003, S. 20) sprechen in diesem Zusammenhang von qualitativ-strategischer Betrachtung.

Die Festlegung einer IT-Strategie ist Teilaufgabe der Strategischen Planung. Sie spezifiziert die Informationsinfrastruktur im Hinblick auf die aus der Unternehmensstrategie abgeleiteten langfristigen strategischen IT-Ziele (vgl. Mertens et al., 2004, S. 179 ff.).

Da die später zu wählenden Alternativen sowohl hinsichtlich ihrer Eignung für strategische Fragestellungen, wie etwa dem grundsätzlichen Wunsch nach Anbieterunabhängigkeit oder nach Etablierung offener Standards, als auch hinsichtlich qualitativer Fragen, etwa nach der Interoperabilität, Stabilität oder Sicherheit, geprüft werden müssen, ist es zwingend erforderlich, bereits während der Phase 1 des Entscheidungsmodells, also in der Projektvorstudie, die qualitativen und strategischen Ziele zu benennen. Dies schafft eine Kultur der genauen Zielformulierung und verhindert, dass reine Bauchentscheidungen in die umfangreiche Analyse ungangbarer Projekte münden. Eine genaue Zielformulierung sorgt für einen verbesserten Überblick und die Möglichkeit der Messung des Zielerreichungsgrades nach und während der Durchführung des Projekts.

Mit der Strategischen Betrachtung von OSS (siehe Abschnitt 2.2.1), den Geschäftsmodellen (siehe Abschnitt 2.2.2) und den Rechtsgrundlagen des Einsatzes von OSS (siehe Abschnitt 2.2.3) – und insbesondere im Zusammenhang mit der Softwarequalität und Reife von OSS (siehe Abschnitt 2.3.4) im Rahmen der Betrachtung technischer Grundlagen von Anwendungssystemen (siehe Abschnitt 2.3) – ergeben sich die in 2.2.4 und 2.3.5 dargestellten Konsequenzen für die Formulierung qualitativer und strategischer Ziele.

Als Hilfe zur Formulierung der qualitativen und strategischen Ziele eignet sich ein Blick auf die in Abbildung 3.3 dargestellten fünf wesentlichen Bestandteile der IT-Strategie und deren Zieldomänen.

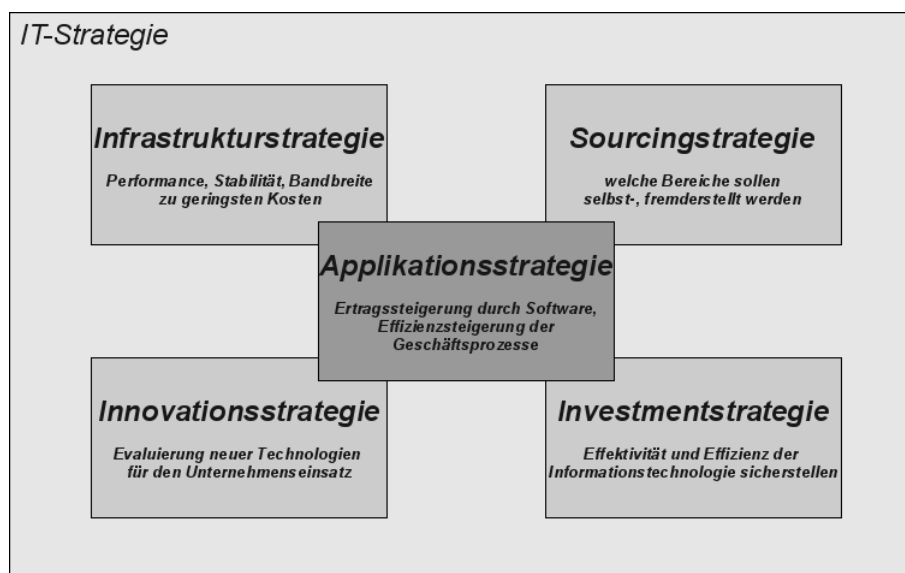


Abbildung 3.3: IT-Strategie - Aufbau nach Schwab (vgl. Schwab, 2007)

Infrastrukturstrategie Im Rahmen der Infrastrukturstrategie werden Basistechnologien, wie Hardware, Netzwerke und Betriebssysteme betrachtet. Ziel dieser Teilstrategie ist die Bereitstellung hoher Rechenleistung, Performance und Bandbreite zu geringen Kosten.

Applikationsstrategie Die Applikationstrategie hat die softwaretechnische Unterstützung von Geschäftsprozessen zum Zweck. Dabei kann Software grundsätzlich entweder zur Ertragssteigerung (z. B. CRM) oder zur Effizienzsteigerung des Geschäftsbetriebes (z. B. ERP) eingesetzt werden.

Innovationsstrategie Die Innovationsstrategie hat die Betrachtung von technologischen Neuerungen und die Bewertung dieser Technologien für den Einsatz im Unternehmen zum Zweck.

Sourcingstrategie Die Sourcingstrategie befasst sich mit der Frage, welche IT-Leistungen selbst erstellt werden und welche am Markt beschafft werden sollen.

Investmentstrategie Die Investmentstrategie legt die IT-Investitionen eines Unternehmens fest. Aus den anderen Teilstrategien abgeleitet hat sie die effiziente und effektive IT-Ressourcenverwendung zum Ziel.

3.1.5 Projektvorstudienbericht

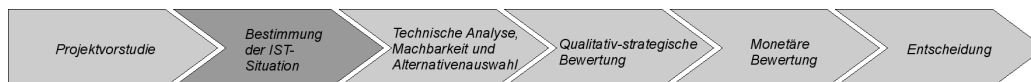
Der Projektvorstudienbericht ist eine schriftliche Zusammenfassung der in Phase 1 durchgeführten Risikoeinschätzungen und Zieldefinitionen. Er dient der Unternehmensleitung als „*Executive Summary*“ und soll neben einer entscheidungsrelevanten Informationsfunktion auch Anreize zur Wahrnehmung der Führungsaufgaben bieten. Das geplante IT-Projekt soll insbesondere im Hinblick auf die strategische Rolle der Informationstechnologie, die laut Ross und Weill (vgl. Ross und Weill, 2006) ebenfalls durch die Unternehmensleitung definiert wird, dargestellt werden und mit Blick auf eine vernünftige Risikoabwägung einschätzbar sein.

Zunächst werden die mit Hilfe der Risikochecklisten (siehe Anhang B.1) erhobenen Risikowerte anhand des in Abbildung B.2 dargestellten Entscheidungsmodells ausgewertet und mit einer entsprechenden schriftlichen Empfehlung versehen.

Insbesondere bei Projekten mit erheblichen Risiken in einzelnen Risikobereichen sind die betroffenen Risiken detailliert darzustellen und Maßnahmen zur Verminderung oder Vermeidung vorzuschlagen. Bei Projekten mit allgemein hohem Risiko muss dieser Sachverhalt entsprechend dargestellt und ein besonderes Augenmerk auf das Risikomanagement bei der späteren Durchführung des Projektes gerichtet werden.

Die qualitativen und strategischen Ziele des IT-Projektes werden im Projektvorstudienbericht zusammenfassend dargestellt und beschrieben. Eine Bewertung erfolgt jedoch erst nach Ermittlung der zu untersuchenden Alternativen im Rahmen der qualitativ-strategischen Bewertung der Phase 4 (siehe Abschnitt 3.4).

3.2 Bestimmung der IST-Situation



Entscheidungsmodell - Phase 2

Die IST-Analyse soll ein möglichst umfassendes Abbild der aktuellen hardware- und softwaretechnischen Infrastruktur des betrachteten Anwendungssystems liefern. Die Datenerhebung wird zu diesem Zweck mit Anleihen aus dem Migrationsleitfaden der „*Koordinations- und Beratungsstelle für Informationstechnik in der Bundesverwaltung (KBSt)*“ (vgl. Langmack, 2006, S. 416 ff.) durchgeführt. Der Migrationsleitfaden schlägt die Unterteilung der Untersuchungsbereiche wie folgt vor:

- Infrastruktur Server
- Infrastruktur Arbeitsplatzcomputer
- Infrastruktur Netz
- Infrastruktur Druck
- Serverdienste
- Office
- IT-Fachverfahren
- Dokumentenvorlagen

Die IST-Analyse laut Migrationsleitfaden erfasst alle relevanten Bereiche einer Migration, ist jedoch sehr auf die öffentliche Verwaltung ausgerichtet. Insbesondere wird im Zusammenhang mit der IT-Unterstützung der Erbringung von Dienstleistungen im öffentlichen Sektor von sogenannten

„*Fachverfahren*“ gesprochen. Es handelt es sich dabei um speziell zugeschnittene Softwarelösungen für die Aufgaben der öffentlichen Verwaltung (vgl. Rüttgers, 2007). Die Verfahrensbeschreibung selbst entspricht dabei dem Geschäftsprozess und das „*Fachverfahren*“ seiner informationstechnischen Unterstützung durch angepasste betriebliche Standardsoftwarelösungen oder entsprechender Individualsoftware, also den expliziten Workflows.

Das Erhebungsmodell eignet sich demnach inhaltlich für alle IT-Projekte, wird aber zu diesem Zwecke neu strukturiert und um einige Details ergänzt. Insbesondere werden die Begriffe „*Fachverfahren*“ und „*Workflow*“ synonym verwendet. Da bereits die Projektvorstudie (vgl. 3.1) einen Überblick über den Umfang des IT-Projektes und damit des betrachteten Anwendungssystems liefert, müssen im Rahmen der Erhebung nur relevante Bereiche untersucht werden. Unnötige Erhebungsaufwendungen können somit bereits an dieser Stelle vermieden werden.

Wenn aus der Abgrenzung des Untersuchungsobjekts (siehe Abschnitt 3.1.2) zum Beispiel hervorgeht, dass für die zentralen Fileserver eines Unternehmens an einem Standort eine Migration / Substitutionsentscheidung getroffen werden soll, ist für dieses Projekt lediglich die Erhebung der „*Serverinfrastruktur*“, „*Serverdienste*“ und „*Netzinfrastruktur*“ wichtig und auch nur derjenigen Systeme, die aktuell an der Fileserverinfrastruktur beteiligt sind.

Da je nach Industriezweig Sicherheit, Stabilität, Performance und Verfügbarkeit hohe Prioritäten haben, müssen bei der Betrachtung etwaige Hot-Standby Systeme (eine redundant vorhandene Infrastruktur, die beim Ausfall sofort den Dienst der ausgefallenen Systeme übernimmt) ebenso wie eventuelle externe Datenspeicher (z. B. Storage Area Networks, Caches und Bandroboter) oder Firewall-, Einbruchssicherungs- und Virencansysteme³ mit erfasst werden. Daher wird das Modell zur IST-Analyse um einen Bereich „*Sonstige Infrastruktur*“ erweitert.

³ Einbruchssicherungssysteme heißen auch Intrusion Detection Systems

Der Tatsache, dass es sich bei einem Anwendungssystem immer auch um den Teil eines Informationssystems handelt, das als umfassenderes Gestaltungsobjekt auch den Menschen bzw. dessen Interaktionen berücksichtigt (vgl. Seibt, 2001, S. 47), wird ebenfalls Rechnung getragen. Bei der Erhebung werden die für den Betrieb und den Support verantwortlichen Know-how-Träger und die Hauptanwendergruppen ebenfalls erfasst. Auf das Administrations- und Supportpersonal kommt im Rahmen eines IT-Projektes in der Regel ein erhöhter Arbeits- und Lernaufwand zu, da es sich im Zuge veränderter Systeme auch mit anderen Architekturkonzepten, Administrationswerkzeugen und Problemlösungsverfahren vertraut machen muss.

Die Teilerhebungen werden entsprechend Abbildung 3.4 drei Erhebungsreichen zugeordnet. Die Erhebung des Betriebspersonals und der Hauptanwender ist dabei als Querschnittsfunktion zu betrachten. Sie soll zum einen Planungs- und Koordinierungshilfe, zum anderen aber auch die Grundlage zur frühen Einbindung der Nutzer sein.

Das Betriebspersonal fungiert zum einen selbst als Nutzer eines Anwendungssystems, zum anderen ist zu erwarten, dass es die Rolle eines Fürsprechers nur dann übernimmt, wenn es frühzeitig von den Vorteilen und dem Erfolg eines Projekts überzeugt ist (vgl. Schröder, 2004, S. 19).

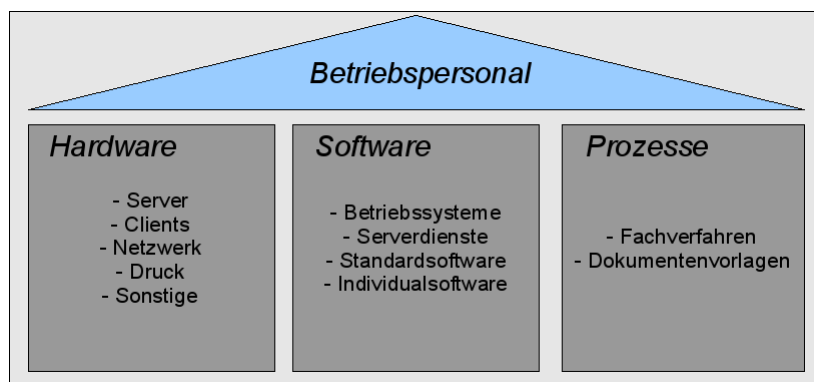


Abbildung 3.4: IST-Erhebungsmodell

Die Erhebungshilfen (siehe Anhang B.3) lehnen sich dabei an die Mustererhebungsbögen der KBSt⁴ an. Sie sind entsprechend dem IST-Erhebungsmodell (siehe Abbildung 3.4) erweitert bzw. strukturell angepasst.

3.2.1 Hardware

Die Erhebung der Hardware soll ein möglichst genaues Abbild der aktuell verfügbaren Infrastrukturkomponenten liefern. Neben der Erhebung technisch relevanter Informationen wie etwa Arbeitsspeichergröße, Prozessoranzahl und -leistung, Festspeichergröße, Übertragungsbandbreite oder Druckkapazität sollen auch Merkmale wie Standort (Rechenzentrum, Rack, Abteilung, Büro), Inventarnummer, Alter und Nutzungsbereich (Abteilung, Person, Arbeitsgruppe) erhoben werden. Durch die Aufnahme der Inventarnummer ist sichergestellt, dass jede Komponente eindeutig bestimmbar und ihr buchhalterischer Wert leicht ermittelbar ist. Insbesondere der Gefahr, dass verknüpfte oder verkettete Infrastrukturkomponenten versehentlich vergessen werden, wird durch Erhebung von „*Verweisen*“ auf andere Komponenten begegnet. Im Ergebnis soll eine Aussage darüber getroffen werden können, ob das Projekt mit der im Bestand befindlichen Hardware durchgeführt werden kann und welche Komponenten ggfs. ausgetauscht bzw. neu beschafft werden müssen. Jeweils der einzelnen Komponente zuordenbare Kosten für Wartung, Ersatz von Verschleißteilen, Verbrauchskosten und sonstige notwendige Investitionen werden als Kosten auf Jahresmittelwertbasis erfasst.

3.2.2 Software

Im Erhebungsbereich Software werden verwendete Betriebssysteme, Serverdienste, Standard- und Individualsoftware erfasst. Im Rahmen der Erfassung

⁴ http://www.kbst.bund.de/cln_046/nn_910242/Content/Wirtschaft__u__Recht/Wirtschaftlichkeit/Migrationshilfen/Mustertabellen/mustertabellen_node.html__nnc=true, besucht am 02.08.2007

der Betriebssysteme (siehe Tabelle B.7) werden die Verweise zu den jeweiligen Hardwarekomponenten ebenso erfasst, wie für die Serverdienste, Standardsoftware und Individualsoftware (siehe Tabelle B.8). Im Gegensatz zum Migrationsleitfaden der KBSt ist Individualsoftware explizit mit zu erfassen. Das macht die Erhebung vollständig und sorgt für einen umfassenden Blick auf die Systemzusammenhänge. Weil Individualsoftware in der Regel dort eingesetzt wird, wo das Leistungspotential der Informationsfunktion am höchsten ist, muss sichergestellt sein, dass diese Teile betrieblicher Anwendungssysteme besonders sorgfältige Beachtung finden. Ein Totalausfall solcher – häufig geschäftskritischen Systeme – ist nicht hinnehmbar und oft sogar existenzbedrohend.

3.2.3 Prozesse

Die informationstechnische Unterstützung der Geschäftsprozesse durch explizite Workflows ist Gegenstand des Erhebungsbereichs „*Prozesse*“. Durch IT-Projekte lassen sich hier weder *schnell* noch *einfach* Kosten sparen, dafür aber nachhaltige Beiträge zum Unternehmenserfolg erzielen. Projekte in diesem Bereich sind um einiges risikobehafteter als reine Infrastrukturprojekte, da die Interaktion des Menschen und vor allem dessen Reaktion auf Änderungen kaum vorhersagbar oder genau zu beschreiben ist. In der Regel entspringen IT-Projekte eher einer Geschäftsprozessoptimierung als rein technologiegetriebenen IT-Vorhaben. Die Maßgabe ist daher vor allem, den für die Nutzer gewohnten Workflow möglichst unverändert beizubehalten. Die sorgfältige Aufnahme des IST-Zustandes erfüllt hierbei eine wichtige, deskriptive Aufgabe. Neben den mittels Customizing (vgl. Görk, 2001, S. 126 f.) von Standardsoftwarelösungen abgebildeten Prozessen und Workflows (siehe Abschnitt 2.1.1) haben Dokumentenvorlagen und Makros eine entscheidende Bedeutung für die individuelle Arbeitsorganisation (vgl. Hertzum, 1999) und werden daher getrennt erfasst.

Dokumentenvorlagen und Makros

Dokumentenvorlagen und Makros bilden, nicht nur im Bereich der öffentlichen Verwaltung, ein eigenes Subsystem. Häufig sind sie zwar in komplexe Standard- oder Individualsoftwareprodukte integriert und somit standardisiert, aber es finden sich auch im Bereich der persönlichen Arbeitsorganisation oder der Arbeitsorganisation von Abteilungen individuelle Dokumentenvorlagen und Makros. In Abhängigkeit vom Geschäftsprozessmodell des Unternehmens und dessen informationstechnischer Unterstützung, können mit ihrer Hilfe Implementierungslücken geschlossen werden. Damit haben sie eine beachtliche Bedeutung für die Arbeitsorganisation. Die Tatsache, dass Informationssysteme häufig eher gewachsen als modelliert sind, sorgt insbesondere in puncto Übersichtlichkeit der Ablage für weitreichende Probleme. Neben der Erhebung möglichst aller Dokumente und Makros und deren Ablageort ist die Einschätzung hinsichtlich ihrer Reimplementierung z. B. bei geplanten Betriebssystem- oder Officemigrationen elementar.

Workflows

Die informationstechnische Unterstützung der Geschäftsprozesse mittels Software zeigt, in welcher Form Informationstechnologie zur betrieblichen Leistungserstellung eingesetzt wird. Langmack (vgl. Langmack, 2006, S. 423) schlägt für die Erhebung der Fachverfahren im öffentlichen Sektor die folgenden „*Informationscluster*“ genannten Erhebungsbereiche vor, die auch diesem Modell in ähnlicher Form zu Grunde liegen.

- Architektur und Nutzer
- Client-/ Serverbetriebs-, Datenbanksysteme und Applikationsserver
- Benutzer-/Rechteverwaltung und Schnittstellen
- Hosting, Verfahrensentwicklung und Ausprägung

- Ausblick, Kosten und Anmerkungen

Im Bereich der öffentlichen Verwaltung sind *Fachverfahren* in der Regel abgeschlossene Softwarelösungen, deren Hauptbearbeitungsobjekt ein „Antrag“ in Dokumentenform ist. Dies gilt für Unternehmen nicht in gleicher Weise, da sie in der Regel neben den Kunden auch mit Lieferanten und Wettbewerbern interagieren, wie das Kräftemodell nach Porter (vgl. Porter, 1999, S. 4) sehr anschaulich zeigt. Dabei hat es ein Unternehmen üblicherweise mit einer Vielzahl betriebswirtschaftlich relevanter Objekte (z.B. Antrag, Auftrag, Reklamation, Kündigung, Bestellung usw.) zu tun. Eine Ausrichtung am Antragsdokument ist daher nicht Gegenstand eines allgemeingültigen Erhebungsmodells.

Der detaillierte Überblick über architekturelle Konzepte, benötigte Serverdienste, Rechtevergabe, Schnittstellen und Support leistet einen Beitrag zur Gesamtdarstellung und macht kritische Komponenten leichter erkennbar. Im Zusammenspiel mit der Erfassung der Software (siehe Abschnitt 3.2.2) entsteht durch die Aufnahme der Arbeitsabläufe (Workflows) innerhalb einer Software oder übergreifend ein genaues Bild der Beschaffenheit und Durchdringung des betrachteten Anwendungssystems.

3.3 Technische Analyse, Alternativenauswahl und Machbarkeit



Entscheidungsmodell - Phase 3

Mit der im Abschnitt 3.2 erstellten Infrastrukturlandkarte sind die Komponenten des betrachteten Anwendungssystems genau beschrieben. Auf

dieser Basis werden nun OSS-Komponenten ermittelt, die aus technischer Sicht prinzipiell geeignet sind, das bestehende Anwendungssystem ganz oder teilweise zu ersetzen. Da mit einer ökonomisch effizienten Analyse nicht alle potentiell bereitstehenden OSS-Produkte geprüft werden können (das OSS-Entwicklungsparadigma kann eine große Produktvielfalt begünstigen), soll bei der technischen Analyse der Betrachtungsrahmen bereits auf drei mögliche Alternativen pro Systemkomponente beschränkt werden.

3.3.1 Auswahlmaßstab erstellen

Um Alternativen zur genaueren Betrachtung selektieren zu können, wird an dieser Stelle ein Maßstab benötigt. Wie bereits in Abschnitt 2.3.4 und Abschnitt 2.3.5 dargestellt, ist die Erhebung produktspezifischer Qualitätsmerkmale ex ante schwierig. Daher wird das von Wichmann (vgl. Wichmann, 2004, S. 24 ff.) vorgeschlagene und in Abschnitt 2.3.4 vorgestellte Verfahren genutzt um AS-Alternativen zu bewerten.

3.3.2 Alternativen suchen

Noch vor wenigen Jahren bestand ein grundsätzliches Problem darin, überhaupt von der Existenz einer geeigneten Open Source Software Kenntnis zu erlangen. OSS-Projekte waren häufig nur den direkt an der Entwicklung beteiligten Personen und Organisationen bekannt, sieht man einmal von bereits früh sehr populärer OSS (Linux, Apache, MySQL) ab.

In den vergangenen Jahren haben sich ein Reihe zentraler Verzeichnisse für Open Source Projektentwicklung, allen voran Sourceforge⁵ (siehe Abschnitt 1.4), etabliert. Mit Hilfe dieser Verzeichnisse lässt sich sehr einfach geeignete OSS finden (vgl. auch Wheeler, 2007).

⁵ andere sind Freshmeat (<http://www.freshmeat.net>), Savannah (<http://savannah.gnu.org>) oder Icewalkers (<http://www.icewalkers.com>)

Die Datenbank kann entweder direkt oder durch eine bereits vorhandene Sortierung in Softwarekategorien durchsucht werden. Für gefundene Projekte werden wesentliche Metadaten (Projektname, Rang, Aktivität, Erstregistrierung, neueste Datei und Downloadzahl) angegeben, die zum Teil direkt Reifemerkmale darstellen (Alter des Projekts, Lizenztyp u. a.). Außerdem ermöglichen Filterfunktionen das Eingrenzen auf bestimmte Eigenschaften (z. B. Lizenztypen).

Es empfiehlt sich bereits zu Beginn der Suche von Anwendungssystemalternativen Mindestanforderungen zu formulieren. Diese müssen mit geringem Aufwand bestimmbar sein und ermöglichen so bereits eine erste Selektion.

Mit Hilfe von Sourceforge lässt sich OSS im gewünschten Anwendungsbereich finden. Auch bereits bekannte OSS-Software lässt sich mit Hilfe von Sourceforge hinsichtlich der bereits genannten Metadaten besser einschätzen bzw. im Verhältnis zu konkurrierenden Projekten bewerten.

Auch eine Webrecherche kann durchgeführt werden. Die Tatsache, dass OSS häufig verteilt über den ganzen Globus entwickelt wird, sorgt dafür, dass eine Vielzahl projektbezogener Websites bzw. Foren sowie Useneteinträge bereits einen guten Überblick über vorhandene OSS gibt. Damit kann ggfs. ein direktes Expertenurteil über einige Indikatoren des Reifemodells nach Wichmann (vgl. Wichmann, 2004, S 24. ff) erfolgen.

Der von Otis et al. (Otis et al., 2007) veröffentlichte Open Source Catalogue stellt eine weitere Quelle dar. Obgleich hier schon sehr detaillierte Aussagen hinsichtlich einer sogenannten „*Enterprise Readiness*“ gemacht werden soll an dieser Stelle lediglich die auf der Produktqualität (siehe Abschnitt 2.3.4) basierende Einschätzung der Reife als Auswahlkriterium herangezogen werden.

Der Vorteil dieser Herangehensweise besteht darin, dass die erfolgreiche Suche bereits eine Reifebewertung beinhaltet und somit zu einer schnelleren Vorauswahl führen kann. Leider beinhaltet der Katalog im Verhältnis zu den

derzeit über 150.000 Open Source Projekten mit 260 Aufgeführten nur eine sehr beschränkte Auswahl.

3.3.3 Alternativen bewerten

Die Reifebewertung ermittelter OSS-Alternativen wird auf Basis des Verfahrens von Wichmann (vgl. Wichmann, 2004, S. 24 ff.) durchgeführt. Die Bewertung der Merkmale erfolgt anhand der vorgestellten Notenskala auf Basis eines Expertenurteils (siehe Abschnitt 2.3.4). Um die ebenfalls mögliche unternehmensspezifische Gewichtung zu realisieren, kann ein Gewichtungsvektor für die drei Erhebungsbereiche (Reife des Produkts, Marktpräsenz der Software, Unternehmensorientierung) festgelegt werden. Formal erfolgt die Zielgewichtung durch die Multiplikation der Nutzenwerte u_{ip} mit positiven Gewichten g_p ($p = 1, 2, \dots, r$). Die Addition der gewichteten Nutzenwerte ergibt die Bewertung einer Alternative mit $\Phi(a_i) = \sum_p g_p u_{ip}$ (vgl. Bamberg und Coenenberg, 2006, S. 56).

Otis et al. (vgl. Otis et al., 2007, S. 10) bewerten die Reife ebenfalls anhand einer fünfstufigen Skala und kann daher direkt mit dem auf Basis des Modells von Wichmann (vgl. 2004, S. 24 ff.) ermittelten Wert verglichen werden. Um eine effiziente Durchführung von weiteren entscheidungsvorbereitenden Schritten zu gewährleisten, sollen nach dem Maximalwertprinzip höchstens drei Alternativen zur weiteren Beurteilung ausgewählt werden.

Auf die Analyse der produktbezogenen Qualität (z.B. Softwaretest, Code Review, Formale Verifikation u. a.) soll zunächst verzichtet werden. Entsprechende Tests sind Bestandteil der Pilotprojekte bzw. „*Proof of Concept*“ und ergeben sich z. B. aus dem notwendigen Vergleich mit aktuell eingesetzten Anwendungssystemen.

3.3.4 Technische Analyse und Machbarkeit

Die Definition und Durchführung geeigneter Pilotprojekte ist für eine letztliche Entscheidung über eine Migration oder Substitution unbedingt erforderlich. Zum einen kann damit sichergestellt werden, dass die geplante Qualität (siehe Abschnitt 2.1.1) auch erreicht werden kann, zum anderen liefern Pilotprojekte Erkenntnisse über die Erreichbarkeit strategischer Ziele und die Machbarkeit mit den zur Verfügung stehenden Ressourcen (vgl. Wichmann, 2004, S. 64).

Zumindest bei Projekten, in denen viele Anwender unmittelbar durch Änderungen betroffen sind (z. B. Umstellung des Client-Betriebssystems), entsteht erst durch die Interaktion mit den am Pilotprojekt beteiligten Anwendern die Möglichkeit, Schulungsbedarf für das unternehmensweite Roll-out zu erkennen, entsprechend zu planen oder ggfs. anzupassen.

Pilotprojekte verringern bei handhabbarer „unkritischer“ Masse das Gesamtprojektrisiko, weil sie zum einen mit erheblich geringerem Mittelaufwand als das betrachtete Gesamtprojekt durchgeführt werden können und zum anderen auch Kosten, insbesondere für Schulungen und Anpassungsaufwände sowie nicht direkt ausgabenwirksame Einarbeitungszeit und Selbsthilfe, einschätzbar machen. Auch das zeitweise Nebeneinander alter und neuer Software lässt sich mit ihrer Hilfe testen. Wichmann (vgl. Wichmann, 2004, S. 65 f.) nennt sechs Planungsziele für die Auswahl geeigneter Piloten. Neben Umfang, Laufzeit und Komplexität sind das Zielkongruenz, Sichtbarkeit (im Sinne produktiver Pilotergebnisse) und aufgeschlossene Pilotnutzer.

Die Dimensionierung eines geeigneten Pilotprojektes hängt stark von der Projektvorstudie (siehe Abschnitt 3.1) und der Bestimmung des IST-Zustandes (siehe Abschnitt 3.2) ab. So kann im Grenzfall (zum Beispiel bei der Substitution von Fileservern) das Aufsetzen eines Servers mit entsprechender OSS und die Durchführung eines geeigneten Performancetests genügen, während insbesondere bei Migrationen (siehe Abschnitt 2.3.1) genaue Migrationspfade

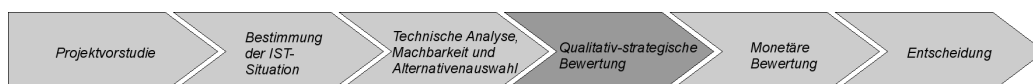
festgelegt, getestet und gegebenenfalls angepasst werden müssen. Bei direkter Beteiligung von Anwendern müssen darüber hinaus Schulungskonzepte erarbeitet und Terminkoordination betrieben werden.

Der Umfang eines Pilotprojektes steigt zwangsläufig auch mit der Komplexität der Einführung (siehe Abschnitt 2.3.1) und der Zahl direkt beteiligter Anwender. Die Ergebnisse eines Piloten müssen Aufschluss darüber geben, ob eine ausgewählte OSS die technischen Anforderungen erfüllt und welche technischen und organisatorischen Probleme bei einem unternehmensweiten Roll-out zu erwarten sind.

Insbesondere muss dem Entscheider nach Durchführung des Pilotprojektes bekannt sein, ob die Mindestanforderungen erreicht sind bzw. welcher Aufwand zur Erreichung entsteht. Je nach Umfang des Pilotprojektes ist dazu ein detaillierter Anforderungskatalog auf Basis eines FCM-Modells zu erstellen und mittels geeigneter Analyseverfahren (i. d. R. Testen der Software) (vgl. Mellis, 2001, S. 422 f.) nach der Alternativenauswahl, also ex post zu ermitteln.

Entwicklungsaufwand kann neben den Schulungskosten und Opportunitätskosten des zeitweiligen Produktivitätsverlust in der Höhe relativ genau erfasst werden. Dadurch ist eine nahezu vollständige Kostenangabe möglich, welche die Datenbasis für die monetäre Bewertung (siehe Abschnitt 3.5) verbessert.

3.4 Qualitativ-strategische Bewertung



Entscheidungsmodell - Phase 4

In der Projektvorstudie (siehe Abschnitt 3.1) wurden u. a. die qualitativen und strategischen Ziele in Übereinstimmung mit der IT-Strategie benannt (siehe Abschnitt 3.1.4) und im Projektvorstudienbericht (siehe Abschnitt 3.1.5) schriftlich erfasst.

Um eine fundierte Entscheidung treffen zu können, müssen daher Kriterien definiert werden, die den Nutzen von Alternativen im Hinblick auf die formulierten qualitativen und strategischen Ziele bewertbar machen.

Das Erreichen unterschiedlicher qualitativ-strategischer Teilziele ist in der Regel ungleich wichtig, daher sollte mit Aufstellung der Kriterien eine Gruppierung und Gewichtung der Kriterien erfolgen.

Im entwickelten Entscheidungsmodell wird die Nutzwertanalyse zum Vergleich der Alternativen herangezogen. Zum einen hat sie sich in der Praxis der europäischen Wirtschaft bei einschlägigen Migrationsfallbeispielen bewährt, zum anderen ist sie Bestandteil des von der KBSt veröffentlichten WiBe-Verfahrens⁶ und dessen Softwareimplementierung WiBe 4.0 (siehe Abschnitt 2.4.3), das auch im Anwendungsbeispiel des Entscheidungsmodells verwendet wird.

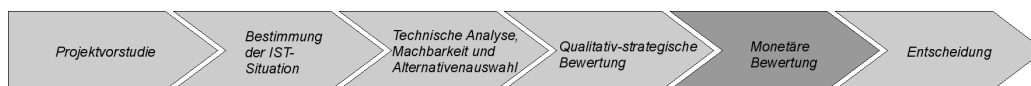
Da die Nutzwertanalyse (siehe Abschnitt 2.4.2) die qualitativ-strategische Betrachtung der in Abschnitt 2.4.3 vorgestellten Wirtschaftlichkeitsbetrachtung WiBe abdeckt, sind zunächst die Zielkriterienbestimmung und deren Gewichtung durchzuführen. Weil im WiBe-Verfahren die Betrachtung der qualitativen und strategischen Faktoren in den Bereichen WiBe Q, WiBe D und WiBe E erfolgt (siehe Abschnitt 2.4.3) muss zunächst bestimmt werden, ob externe Faktoren (WiBe E) und/oder Dringlichkeitsfaktoren (WiBe D) für das konkrete Projekt überhaupt relevant sind. Durch den allgemeingültigen Anspruch des hier vorgestellten Entscheidungsmodells werden die qualitativen und strategischen Ziele nicht notwendigerweise den Vorgaben des gene-

⁶ http://www.kbst.bund.de/c1n_046/nn_836962/Content/Wirtschaft__u__Recht/Wirtschaftlichkeit/wirtschaftlichkeit__node.html__nnn=true, besucht am 8.7.2007

rellen Kriterienkataloges entsprechen. Das Anpassen des Kriterienkataloges auf die besonderen Projektbedürfnisse ist daher ebenfalls Bestandteil der qualitativ-strategischen Betrachtung.

Im Anschluss an Anpassung oder Neuanlage des entsprechenden Kriterienkataloges und der Formulierung der aus den qualitativen und strategischen Zielen (siehe Abschnitt 3.1.4) abgeleiteten Kriterien (siehe Abschnitt 2.4.2) können diese Kriterien mit Hilfe der Softwareimplementierung WiBe 4.0 bewertet werden.

3.5 Monetäre Bewertung



Entscheidungsmodell - Phase 5

Die Monetäre Bewertung der Alternativen erfolgt auf Basis der in Abschnitt 2.4.4 durchgeführten kritischen Würdigung der vorgestellten Verfahren ebenfalls nach WiBe.

Zunächst ist zu diesem Zweck wiederum der generelle Kriterienkatalog im Bereich WiBe KN anzupassen bzw. ein neuer Kriterienkatalog festzulegen. Aus den in der Projektvorstudie definierten Zielen, der Abgrenzung des Projektrahmens, der anschließenden Bestimmung der IST-Situation und darauf folgender Alternativauswahl und -analyse ergeben sich zum einen die Kosten eines im Einsatz befindlichen AS als auch die Kosten für die AS-Alternativen. Anhand erster Pilotprojekte lassen sich notwendige Entwicklungsaufwände ebenso bestimmen wie die zu erwartenden Betriebskosten der AS-Alternativen. WiBe ermöglicht darüber hinaus auch die Aufzinsung bereits durch die Projektvorstudie selbst anfallender Kosten und deren Aufnahme in das Kosten-/Nutzenkalkül.

Nachdem der Kriterienkatalog entsprechend angepasst wurde, kann auf Basis bereits erhobener Kosten-/Nutzenposition die monetäre Bewertung erfolgen. Kosten/Nutzen, die erst zu einem späteren Zeitpunkt des Projektverlauf deterministisch bestimmbar sein werden, können im Rahmen des Grobkonzeptes (siehe Abschnitt 2.4.3) zunächst geschätzt werden.

Zur Durchführung der Bewertung wird ebenfalls die Softwareimplementierung WiBe 4.0 verwendet.

3.6 Entscheidung



Entscheidungsmodell - Phase 6

Für das Entscheidungsmodell wird der abgezinste⁷ und mit Risikozuschlag versehene Kapitalwert (Risikowert WiBe KN/R) mit der durch Nutzwertanalyse ermittelten qualitativ-strategischen Bewertung (WiBe Q) durch Verwendung der Portfoliotechnik visualisiert. Die so dargestellten Ergebnisse werden in jeweils zwei Abstufungen, also in vier Quadranten, unterteilt. Für jeden dieser Quadranten wird eine Normhandlungsstrategie wie folgt festgelegt:

Alternativen mit einem hohen relativen Risikowert (monetäre Vorteilhaftigkeit) und hohem Nutzwert sind generell vorteilhaft. Alternativen mit hohem Nutzwert bei gleichzeitig niedrigem relativen Risikowert (monetäre Unvorteilhaftigkeit) sollten, wenn nicht von einer anderen Alternative dominiert, nochmals kritisch hinsichtlich ihrer Zukunftspotentiale geprüft werden. Ein Blick in die Kostenkalkulation macht darüber hinaus Kostentreiber erkennbar. Gegebenenfalls können durch leichte Konfigurationsänderungen erheblich Kosten gespart und die Alternative so in eine günstigere, weil gesamtvorteilhafte,

⁷ es wird der aktuelle Kapitalmarktzins verwendet

transformiert werden. Für Alternativen mit geringem Nutzwert bei gleichzeitig hohem Risikowert müssen Zukunftsperspektiven geprüft werden. Es hat ferner eine Abwägung der Gesamtprojektrisiken gegen den monetären Vorteil zu erfolgen. Alternativen im verbleibenden Quadranten sollten nicht umgesetzt werden, da sie durch relativ geringe Erfüllung qualitativ-strategischer Zielsetzungen und niedrigen relativen Risikowert gekennzeichnet und so im Hinblick auf langfristige strategische Ziele untauglich sind. Bei hoher Konzentration der Alternativen in einem Nutzwertbereich sollten die Messkriterien und besonders deren Gewichtung erneut geprüft und ggfs. angepasst werden, um qualitativ-strategische Unterschiede zwischen den Alternativen eindeutiger zu differenzieren.

Basierend auf dieser Visualisierung ist ein Entscheider oder Entscheidungstab in der Lage, direkt über die Durchführung des IT-Projekts zu entscheiden. Durch die detaillierte Erfassung aller projektrelevanter Informationen mit der WiBe-Software lassen sich an dieser Stelle auch gezielt Maßnahmen zur Änderung des Projektumfangs und/oder -konfiguration ergreifen⁸, um so Einfluss auf die Wirtschaftlichkeitsbewertung zu nehmen.

⁸ ist Führungsaufgabe

Kapitel 4

Anwendungsbeispiel des Entscheidungsmodells

4.1 Kurzvorstellung der Placeholder GmbH

Die Placeholder GmbH mit Sitz in Holthausen im Süden Dortmunds ist ein kleines 1988 gegründetes, inhabergeführtes Handelsunternehmen der Land- und Baumaschinenindustrie. Die Placeholder GmbH besitzt langjähriges Fachwissen im Bereich „*Zentralschmiertechnik*“ und bietet ihren Kunden Ersatzteile zur Wartung dieser Anlagen. Im Wesentlichen erwirtschaftet die Placeholder GmbH ihren Umsatz von ca. 8,55 Mio Euro jährlich mit dem Handel von Armaturen und Druckleitungen. Daneben hat die Placeholder GmbH vor elf Jahren in Zusammenarbeit mit einem Lieferanten ein neuartiges Reifenpannenschutzmittel für Land- und Baumaschinen entwickelt, mit dem sie im Wirtschaftsjahr 2007 einen Umsatzanteil von circa 35% erzielte. Das Unternehmen hat vier Angestellte. Zwei der Angestellten arbeiten im Lager und sind mit der Kommissionierung, Warenannahme und Kundenreparaturen beschäftigt. Der Inhaber und ein weiterer Mitarbeiter sind im Bereich Wareneinkauf und -verkauf tätig und ein weiterer Mitarbeiter in der Neukundenakquisition. Daneben beschäftigt die Firma projektbezogen weitere Personen im Bereich der Kundenakquisition. Die Frau des Firmeninhabers ist für die Buchführung des Unternehmens verantwortlich. Seit einiger Zeit wird bei der Placeholder GmbH über den Ersatz der im

Einsatz befindlichen Branchenlösung „Primus“ diskutiert. Zum einen ist das Anwendungssystem sehr instabil, zum anderen ist es auf Handwerksbetriebe zugeschnitten und nicht auf Handelsbetriebe wie die Placeholder GmbH. Dieser Umstand resultiert aus einer Fehlberatung beim Kauf dieser Branchenlösung. Mit dem hier vorgestellten IT-Projekt soll daher geklärt werden, ob eine Migration¹ zu einer für einen Handelsbetrieb geeigneten ERP-Lösung (Enterprise Resource Planning) für das Unternehmen vorteilhaft ist.

4.2 Projektvorstudienbericht

Ziel der Projektvorstudie ist die Abgrenzung des Projekts (siehe Abschnitt 3.1.2), das Erfassen der Projektrisiken (siehe Abschnitt 3.1.3) sowie der zur Verfügung stehenden Ressourcen (siehe Abschnitt 3.1.1). Darüber hinaus werden in der Projektvorstudie die qualitativen und strategischen Projektziele (siehe Abschnitt 3.1.4) benannt. Die Ergebnisse werden durch den Projektvorstudienbericht schriftlich erfasst (siehe Abschnitt 3.1.5).

Der Placeholder GmbH ist es durch verstärkte Investitionen in Messeauftritte und in der direkten Neukundenaquisition gelungen, ihren Kundenstamm erheblich zu vergrößern. Dies schlug sich in einem 70%igen Umsatzwachstum zwischen 2005 und 2006 nieder. Auch dieses Jahr hat die Placeholder GmbH nach Hochrechnung des ersten Halbjahres das 20%ige Wachstumsziel bereits wieder deutlich übertroffen.

Als wesentlich für die Kunden haben sich die schnelle Verfügbarkeit der Waren und die niedrigen Preise sowie die hohe Qualität der Leistungserbringung und Kundenkommunikation herausgestellt. 2007 hat die Placeholder GmbH zur Transparenzsteigerung gegenüber ihren Kunden und Lieferanten eine Zertifizierung nach DIN EN ISO 9001:2000 durchführen lassen. In diesem Zusammenhang wurde die Erbringung sämtlicher Dienstleistungen und

¹ mindestens Kunden- und Artikelstammdaten sollen übernommen werden

Warenlieferungen mit der höchstmöglichen Kundenzufriedenheit als strategisches Unternehmensziel festgeschrieben.

Die Geschäftsleitung der Placeholder GmbH verspricht sich von dieser Ausrichtung auf Kundenbedürfnisse langfristig zunehmende Umsätze auch aus dem Ausland. Mit der langfristigen Unternehmensstrategie soll die weltweite Ausbreitung des Kundenkreises vorangetrieben werden. Die Placeholder GmbH verfügt selbst über kein Know-how zur Durchführung von IT-Projekten. Auch Anwendungswissen ist in der Vergangenheit vorwiegend autodidaktisch erworben worden. Die Geschäftsleitung ist sich der strategischen Bedeutung der Informationsinfrastruktur bewusst. Insbesondere durch die Instabilität der eingesetzten Branchenlösung „*Primus Handwerk*“² sind in der Vergangenheit Produktivitätsausfälle in für das Unternehmen unangenehmer, jedoch nicht existenzbedrohender Höhe aufgetreten, deren Eintrittswahrscheinlichkeit für die Zukunft unbedingt reduziert werden soll.

Besonders die Funktionalität der Branchenlösung in Verbindung mit dem Kommunikationsmedium „*Internet*“ ist sehr eingeschränkt. Die Branchenlösung stellt neben einer sehr einfachen integrierten E-Mail-Funktionalität, welche der Hersteller nicht für den Produktivbetrieb empfiehlt, eine kostenpflichtige MAPI-Schnittstelle zu MS-Outlook bereit, welche die Placeholder GmbH allerdings nicht nutzt. Beide Lösungen sind aus Sicht der Placeholder GmbH nicht optimal. Insbesondere möchte die Placeholder GmbH möglichst flexibel hinsichtlich einer Anbindung des Warenangebots an einen Internetshop sein.

Daneben werden die in Microsoft WindowsTM integrierten Telefoniedienste (CTI) über eine systemintegrierte Telefonanlage zur Verfügung gestellt und teilweise genutzt. Insgesamt soll durch die Verbesserung der hauseigenen Informationsinfrastruktur eine spürbar bessere Kommunikationsqualität gegenüber den Kunden nachhaltig sichergestellt werden. Daneben sollen nach Möglichkeit die Darstellbarkeit der Weiterverarbeitung von Halb-

² mittlerweile durch SAGE (ERP-Marktführer im Bereich KMU) übernommen

fertigerzeugnissen zu Fertigerzeugnissen abgebildet sowie eine Lagerbuchhaltung ermöglicht werden. Die derzeitige Lösung bietet hierzu nur unzureichende Möglichkeiten. Das Erfassen der Artikelgewichte ist für die Optimierung der Versandkosten ein wichtiger Faktor und ermöglicht über das Wiegen einer ausgehenden Lieferung darüber hinaus eine zusätzliche Qualitätssicherung in der Versandkette. Das Unternehmen hat im vergangenen Jahr ein neues Betriebsgelände und Gebäude im 10 km entfernten Handorf erworben. Allerdings wird derzeit nur der Standort Holthausen genutzt. Da das neu erworbene Betriebsgelände in naher Zukunft zum Hauptlagerstandort werden soll, müssen evaluierte Alternativen zwingend multistandortfähig sein. Dabei soll insbesondere auch die im lokalen Netzwerk (LAN) sehr schlechte Performance der aktuellen Branchenlösung verbessert werden. Ein mobiler Zugang zur Branchenlösung und möglichen Alternativen ist für die Unternehmensleitung ebenfalls wichtig. Zwingende, harte Abhängigkeiten zwischen Hard- und Software bestehen nicht. Im Allgemeinen ist die Budgetaustattung für das geplante Projekt als gut zu bewerten und die Unterstützung durch die Unternehmensleitung ist ebenfalls gut.

Die Vorauswahl der Risikobereiche in der Risikocheckliste ist für den Anwendungsfall der Placeholder GmbH adäquat und wird daher nicht erweitert. Die Durchführung der Risikoanalyse des betrachteten Anwendungssystems mittels Tabelle B.1 ergibt einen Risikomittelwert von 3. Dabei ist das Projekt durch ein relativ hohes mittleres Risiko gekennzeichnet, allerdings gibt es keinen ausnehmend hoch bewerteten Risikoteilbereich. Daher stehen gemäß Abbildung B.2 der weiteren Entscheidungsvorbereitung des geplanten Projekts keine grundsätzlichen Risiken entgegen. Die notwendigen Maßnahmen werden von der Unternehmensleitung grundsätzlich unterstützt. Insbesondere gilt dies für die bei der Einführung notwendige, Bereitstellung personeller Ressourcen (Mitarbeiterschulung, Pilotprojekt) und für die notwendigen Investitionen im Rahmen des Migrationsprojekts.

Wie eingangs erwähnt ist die von der Placeholder GmbH eingesetzte Branchenlösung „*Primus Handwerk*“ eigentlich für den Einsatz in Handwerksbe-

trieben bestimmt. Typische Geschäftsvorfälle von Handels- und Produktionsbetrieben sind nicht oder nur unzureichend umgesetzt. Teilweise wurde das System in der Vergangenheit durch den Hersteller um entsprechende Funktionen (z. B. rudimentäre Lagerbuchhaltung) erweitert oder es wurden mit verfügbaren Mitteln entsprechende Workarounds geschaffen. Gegenstand des betrachteten Projekts ist daher, wie beschrieben, der Wechsel auf eine vollständige ERP-Standardlösung für KMU.

Für die Placeholder GmbH spielt die freie Anpassbarkeit des Quellcodes eine untergeordnete Rolle, da Veränderung am Quellcode des einzuführenden Systems nur im Bedarfsfall durch einen beauftragten Dienstleister durchgeführt werden und von weiteren Rechten als dem einfachen Nutzungs- bzw. Veränderungsrechts kein Gebrauch gemacht werden wird.

Wie jedes zur ordentlichen Buchführung verpflichtete Unternehmen hat sich die Placeholder GmbH an die GoB, und speziell die GoBS, sowie die GDPdU zu halten. Daher wird insbesondere der Datenhaltung ein hoher Stellenwert beigemessen. Durch den erweiterten Handel mit den USA und dem dort gültigen Haftungs- und Gewährleistungsrecht sowie der Mitgliedschaft in der Kreditversicherungsgesellschaft Creditreform ist für das Unternehmen vor allem die lückenlose Nachweisbarkeit der Geschäftsvorfälle von vorrangigem Interesse.

Folgende qualitative und strategische Ziele sowie Messkriterien wurden für das Projekt festgelegt:

Verbesserung der Funktionalität (Kriterien: siehe Tabelle 4.2)

Verbesserung der Datenhaltung (Kriterien: Softwarequalität Datenhaltung, Entwicklungsmodell und Architektur)

Erhaltung des Supportlevels (Kriterien: Verfügbarkeit Dienstleister, Wartungsverträge, Installation zusätzlicher Clients)

Verbesserte Informationsverfügbarkeit (Kriterien: Dokumentenmanagement, Historie, Kommunikationsdienste)

Verbesserung der Workflows (Kriterien: Zeitaufwand der Workflows, Verbesserung der Dateneingabequalität)

4.3 Bestimmung der IST-Situation

Mit Bestimmung der IST-Situation entsteht ein Überblick über das zu ersetzende Anwendungssystem. Durch die Erhebung von Hardware (siehe Abschnitt 3.2.1), Software (siehe Abschnitt 3.2.2) und Prozessen (siehe Abschnitt 3.2.3) wird gewährleistet, dass alle im Rahmen einer Migration/Substitution betroffenen Systemkomponenten berücksichtigt werden. Die Erfassung der Nutzer und des Betriebspersonals ist dabei Querschnittsfunktion für alle anderen Erhebungsbereiche. Kritische Systemkomponenten zur expliziten Prüfung in einem Pilotprojekt werden ebenso ermittelt wie Wartungs-, Betriebs- und/oder Migrationskosten. Vor allem kann in Verbindung mit dem Pilotprojekt (siehe Abschnitt 4.4.4) auch ein Urteil über die Weiterverwendbarkeit von Komponenten getroffen werden.

Das betrachtete Anwendungssystem umfasst nahezu die gesamte Hardwareinfrastruktur der Placeholder GmbH. Lediglich ein Kopiergerät und ausschließlich privat genutzte Clientsysteme wurden nicht erfasst. Zwar ist für das beschriebene Anwendungssystem zunächst kein Wechsel des Betriebssystems geplant, dennoch sind die Clientgeräte durch den aktuell installierten „*Primus Client*“ direkter Bestandteil des Anwendungssystems und daher mitzuerfassen.

Die angenommenen Wartungskosten für die Anwendungssystemkomponenten wurden soweit möglich geschätzt oder anhand von Wartungsrechnungen aus vergangenen Perioden fortgeschrieben. Die betrachteten Hardware- und

Softwarekomponenten wurden dazu in den Mustererhebungsbögen im Anhang B dieser Arbeit erfasst. Da die Prozesse und Workflows, wie in kleinen Unternehmen üblich, leider auch nach Durchführung des Qualitätsaudits mit der DIN ISO EN 9001:2000 Zertifizierung nicht hinreichend dokumentiert waren, wurden zur Entscheidungsvorbereitung und Transparenzsteigerung die Wertschöpfungskette des Unternehmens (siehe Abschnitt B.4), der für das Unternehmen elementare Kundenbestellprozess (siehe Abschnitt B.5) sowie das Organigramm des Unternehmens (siehe Abschnitt B.3) modelliert bzw. dokumentiert.

Insbesondere wurde durch diese Maßnahme eine frühe Beteiligung der Nutzer während der Entscheidungsvorbereitung realisiert, da die Modellierung durch Beobachtung der IST-Situation erfolgte. Mitarbeiter des Unternehmens waren damit direkt an den Modellierungen beteiligt. Reine Office-Dokumentenvorlagen benutzt die Placeholder GmbH zur betrieblichen Leistungserstellung nur in sehr begrenztem Umfang zur Organisation der Warenabholung beim Kunden und zur Beauftragung spezieller Expresslieferungen bei der zuständigen Spedition. Da eine Officemigration vorerst nicht Gegenstand der Planung ist, entstehen hierfür keine Migrationskosten.

4.4 Alternativenauswahl, Technische Analyse und Machbarkeit

An dieser Stelle wird mit Hilfe des Bewertungsmodells nach Wichmann (vgl. Wichmann, 2004, S. 24 ff.) (siehe Abschnitt 2.3.4, 3.3.1) ein Bewertungsmaßstab zur Bewertung der OSS-Alternativen festgelegt und die einzelnen Bereiche (Reife des Projekts, Marktpräsenz der Software und Unternehmenorientierung des Projekts) projektspezifisch gewichtet.

Zur Vorselektion empfiehlt es sich, das Suchfeld entsprechend einzugrenzen. Dazu können bereits Mindestanforderungen formuliert werden, deren

Erfüllung sich einfach bestimmen lässt. Durch das in Abschnitt 3.3.2 beschriebene Vorgehen können anschließend OSS-Alternativen ermittelt und bewertet werden. Nach erfolgter Bewertung werden mit dem Maximalwertprinzip bis zu drei Alternativen ausgewählt und ein Pilotprojekt für jede Alternative durchgeführt. Die Ergebnisse der Pilotprojekte (siehe Abschnitt 3.3.4) gehen dabei direkt in die qualitativ-strategische sowie monetäre Bewertung ein.

4.4.1 Erstellen des Auswahlmaßstabes

Für das Projekt der Placeholder GmbH wurde ein abweichender Bewertungsvektor zur Bestimmung der Einsatzreife festgelegt. Die Einsatzreife wird, wie in Abschnitt 3.3.3 dargestellt, auf Basis der Reife des Projekts, Marktpräsenz der Software und Unternehmensorientierung mit dem individuellen Bewertungsvektor (siehe Formel 4.1) für die OSS-Alternativen bestimmt. Damit soll insbesondere berücksichtigt werden, dass die Placeholder GmbH vor allem ein aus technischer Sicht reifes OSS-Produkt mit guter Unternehmensorientierung einsetzen möchte, während die Marktpräsenz weniger bedeutend ist. Im Urteil der Placeholder GmbH sind sowohl Reife des Projekts als auch Unternehmensorientierung doppelt so wichtig wie die Marktpräsenz. Damit ergibt sich der normierte Bewertungsvektor 4.1.

$$\vec{b} = \begin{pmatrix} 0.4 & 0.2 & 0.4 \end{pmatrix} \quad (4.1)$$

4.4.2 Alternativensuche

Die zu suchende OSS ERP-Alternative muss mindestens folgende Module beinhalten:

1. Finanzbuchhaltung

2. Lagerbestandsführung
3. Einkauf
4. Verkauf
5. CRM

Die Suche nach dem Stichwort „ERP“ liefert bei Sourceforge 328 Suchergebnisse³. Eine darauffolgende Eingrenzung der Projekte nach folgendem Filter

1. Anmeldung des Projekts bei Sourceforge bis 10.08.2006
2. Aktivitätsindex $\geq 95\%$
3. letzter Upload spätestens 10.07.2007

liefert nur noch neun Projekte. Von diesen neun Projekten wurden die beiden mit dem besten Rang auf Sourceforge selektiert, welche die Mindestanforderung erfüllten (Openbravo, Opentaps). Der Sourcecode des Projektes TinyERP wird nicht mehr über Sourceforge verwaltet; das Projekt ist dennoch sehr weit verbreitet und erfüllt ebenfalls die Mindestanforderungen. Das bekannteste OSS ERP-System ist Compiere. Auf Grund fehlender Ressourcen und Meinungsverschiedenheiten mit der Community spaltete sich Ende 2006 das Projekt Adempiere ab. Die Zahl der Subversion-Transaktionen (SVN) auf Sourceforge zeigt, dass Adempiere seitdem wesentlich aktiver entwickelt wird als Compiere (siehe Abbildung B.2). Um Adempiere hat sich eine sehr stabile Community entwickelt, die das Projekt sehr aktiv vorantreibt und nahezu vollständig aus ehemaligen Nutzern und Dienstleistern von Compiere besteht. Adempiere wurde seit seiner Anmeldung auf Sourceforge über 90.000 mal heruntergeladen⁴ und ist auf Rang 11 aller bei Sourceforge gelisteten Projekte derzeit das populärste Open Source ERP-System. Adempiere wird daher an Stelle von Compiere bewertet.

³ Stand 9.8.2007, 20:30 Uhr

⁴ Sourceforge Stand 22.08.2007

4.4.3 Alternativenbewertung

Die in Abschnitt 4.4.2 ermittelten und vorselektierten Alternativen erhalten mit dem spezifischen Bewertungsvektor der Placeholder GmbH (siehe Gleichung 4.1) die in Tabelle 4.1 dargestellten Bewertungen.

| Name des ERP-Systems | Reife des Projekts | Marktpräsenz | Unternehmensorientierung | projekt-spezifische-Gewichtung |
|----------------------|--------------------|--------------|--------------------------|--------------------------------|
| Adempiere | 5 | 4 | 3 | 4 |
| Openbravo | 4 | 4 | 3 | 3,6 |
| TinyERP | 4 | 3 | 3 | 3,4 |
| Opentaps | 5 | 3 | 2 | 3,4 |

Tabelle 4.1: Bewertung der ERP-Alternativen mit dem Bewertungsvektor der Placeholder GmbH

Die Berater von Optaros (vgl. Otis et al., 2007, S. 38 f.) empfehlen lediglich den Einsatz von Adempiere/Compiere und Opentaps nach Durchführung eines detaillierten „Proof of Concept“ für einen Produktiveinsatz. Einen positiven Trend erkennt Optaros bei Openbravo und TinyErp. In dieser Arbeit werden die OSS-Alternativen „Adempiere“, „Openbravo“ und „TinyErp“ exemplarisch analysiert. Die Alternative Opentaps wird nicht weiter analysiert, da sie mit 3,4 dieselbe Bewertung wie TinyERP erhielt und an dieser Stelle maximal drei Alternativen weiter betrachtet werden (siehe Abschnitt 3.3.3).

4.4.4 Technische Analyse und Machbarkeit

Die Primus Branchenlösung verfügt lediglich über eine „Transaktionsengine“ aber nicht über eine zum SQL-Standard konforme Datenhaltung. Eine harte Migration (siehe Abschnitt 2.3.1) wurde in Absprache mit dem Kunden bereits im Vorfeld verworfen. Sämtliche Daten sind lediglich aus manuell bedienbaren Exportschnittstellen zu exportieren. Zur Entwicklung automatischer Migrationskripte stehen keine geeigneten Schnittstellen zur Verfügung. Daher ist eine automatisierte Migration kaum gangbar.

Bei der Migration sind darüber hinaus weitreichende Änderungen an der Datenstruktur zu erwarten. Spezielle Funktionen der „Handwerkslösung“ werden nach Einführung einer ERP-Lösung nicht mehr benötigt (z. B. Funktionen zur Anlage von Aufmaßen). Entsprechende Daten werden demnach nicht migriert. Zur Erreichung des Konsolidierungs- und Migrationsziels wurde beschlossen, nur die wesentlichen Stammdaten zu migrieren. Insbesondere sind das:

1. Kundenstammdaten
2. Lieferantenstammdaten
3. Artikelstammdaten

Anhand der definierten qualitativen und strategischen Ziele sowie der Dokumentation und Beobachtung des im Einsatz befindlichen Anwendungssystems wurden funktionale Anforderungen der einzuführenden AS zur Prüfung im Rahmen des Pilotprojekts abgeleitet (siehe Tabelle 4.2). Die Ergebnisse aus dem Pilotprojekt werden mit der qualitativ-strategischen Bewertung der Alternativen (siehe Abschnitt 4.5) zum Bewertungsbestandteil.

| Funktionale Anforderungen | | | |
|----------------------------------|------|------|------|
| | Muss | Soll | Kann |
| Deutsche Lokalisierung möglich | X | | |
| Kundenkategorien | X | | |
| versch. Währungen | X | | |
| versch. Mehrwertsteuersätze | X | | |
| versch. Zahlungsziele | X | | |
| Gesprächspartnererfassung | X | | |
| Mehrpriessfähigkeit | X | | |
| E-Mail-Funktion | | X | |
| Teillieferung | | X | |
| Mehrlagerfähigkeit | | X | |
| Produktion / Stücklisten | | X | |
| Anbindung Internetshop | | | X |
| DATEV Export | | | X |
| SKR03 Kontenrahmen | | | X |
| Artikelgewichte | | | X |
| Konsignationslager | | | X |

Tabelle 4.2: Funktionale Kriterien für AS-Alternativen

Auf Grund der Tatsache, dass die Placeholder GmbH nur über einen sehr begrenzten Mitarbeiterstab verfügt und für das Pilotprojekt nur ein sehr beschränkter Zeitraum zur Verfügung steht, wird das Pilotprojekt bei dieser Beispielanwendung gänzlich ohne Beteiligung der Mitarbeiter und in verkürzter Form durchgeführt. Im Wesentlichen werden mit dem Pilotprojekt die Umsetzung der funktionalen Kriterien (siehe Tabelle 4.2) und die prinzipielle technische Durchführbarkeit der Migration ermittelt. Auch die Darstellbarkeit der Organisationsstruktur (durch Rollen-/Rechtevergabe) und des Kundenbestellprozesses (siehe Abschnitt B.5) soll geprüft werden. Darüber hinaus soll das Projekt Aufschluss über die technischen Anforderungen an die Einführung geben.

Ein im Rahmen einer Diplomarbeit durchgeführter Vergleich von OSS ERP-Systemen liefert wesentliche Grundinformationen über die technischen Eigenschaften von TinyERP und Adempiere/Compiere (vgl. Herzog, 2006, S. 44 ff., S. 63 ff.). Sowohl Openbravo als auch Adempiere beruhen ursprünglich auf der Codebasis von Compiere. In Verbindung mit den Technischen Dokumentationen⁵⁶⁷ aller drei OSS-Alternativen ergibt sich die in Tabelle 4.3 dargestellte technische Übersicht.

Für das Pilotprojekt stehen für Openbravo und Adempiere VMware Appliances zur Verfügung, die mit einem VMware Player genutzt werden können. Der Aufbau einer Testinstallation für TinyERP erfolgte auf folgender Systemkonfiguration.

Betriebssystem Server: Gentoo Linux, Kernel 2.6.18-gentoo-r6

Datenbankserver: Postgresql-8.0.13

Applikationserver: Python 2.4.4

⁵ <http://tinyerp.org/wiki/index.php/TechnicalDocumentation/HomePage> (besucht am 12.08.2007)

⁶ http://www.adempiere.com/wiki/index.php/Developer%27s_Technical_Manual (besucht am 21.08.2007)

⁷ http://wiki.openbravo.com/wiki/index.php/Main_Page (besucht am 21.08.2007)

| Übersicht technischer Details | | | |
|-------------------------------|---|---------------------------------|---------------------------------|
| | TinyERP | Adempiere | Openbravo |
| Architektur | 3-tier, rich GUI-Client, Webclient | 2,3-tier Fat Client | 3-tier, Webclient |
| Datenbank | Postgresql | Oracle, Postgresql | Oracle, Postgresql |
| Betriebssystem Server | Linux, Windows | Linux, Windows | Linux, Windows |
| Betriebssystem Client | Linux, Windows, MacOS X | Linux, Windows, MacOS X | egal |
| Customizing | XML Formulare und XML basierte Workflows, Reports | Application Dictionary, Reports | Application Dictionary, Reports |
| Applikationsserver | Phyton | JBoss | JBoss |
| Schnittstellen Server | XML-RPC, eCommer-ce | eCommerce | eCommerce |
| Schnittstellen Client | Excel, CSV, PDF, OpenOffice | Excel, CSV, PDF | Excel, CSV, PDF |

Tabelle 4.3: Übersicht technischer Details

Betriebssystem Client: WindowsTM XP Prof. und Ubuntu Linux 7.04 Feisty

Applikationsversion: Tinyerp Version 4.0.3

Durch das Pilotprojekt konnte die verhältnismäßig einfache Installation der Systeme belegt werden. Allerdings ist das Aufsetzen einer geeigneten Produktivumgebung für Adempiere und Openbravo aufwendiger, da neben dem Applikationsserver und der Datenbank entsprechende Java Runtime Environments (JRE) installiert werden müssen (siehe Tabelle 4.8).

Konzeptuell ist die Trennung von Datenmodell, Programmsteuerung und Präsentation durch den MVC-Ansatz (Model, View, Controller) in allen drei OSS-Alternativen umgesetzt. Die Anwendungssysteme sind damit leicht erweiterbar. Adempiere und Openbravo ermöglichen das Customizing über ein integriertes Application Directory. Im Falle von TinyERP kann die Modellierung von Geschäftsprozessen mit einem eigenen Workflowdesigner umgesetzt werden. Zum Customizing der Anwendung muss für die Placeholder GmbH daher keine Zusatzentwicklung durchgeführt werden. Die bereits durch die Studie von Herzog (vgl. Herzog, 2006) bekannte Anpassbarkeit der Systeme konnte mit dem Pilotprojekt verifiziert werden.

Der Quellcode von TinyERP ist kaum dokumentiert, durch die Struktur und das Entwicklungskonzept allerdings gut lesbar. Die Quellcode Dokumentation für Adempiere und Openbravo ist besser als die von TinyERP. Der Anfangsaufwand für eventuelle Zusatzentwicklungen ist auf Grund des MVC-Ansatzes in alle drei Fällen gering. Gute Entwicklerdokumentationen stehen in allen drei Fällen zur Verfügung.

Im Hinblick auf die Steigerung der Datenqualität und Datenverfügbarkeit bringt die Datenhaltung in einer Postgresql- oder Oracle-Datenbank erhebliche Vorteile (Integritätssicherung, „Point in time“-Recovery u. a., Skalierung) gegenüber der jetzigen Lösung. Insbesondere im Zusammenhang mit gesetzlichen Richtlinien (GoB, GoBS und GDPdU) können in Zukunft benötigte Reports und Schnittstellen umgesetzt werden. Die Datenhaltung in einer relationalen Datenbank ermöglicht auch die Entscheidungsunterstützung der Geschäftsleitung durch Business Intelligence.

TinyERP bietet als einzige Alternative den Zugriff sowohl über eine Clientapplikation als auch einen Webclient an. Der Webclient ist allerdings in einem sehr frühen Entwicklungsstadium und damit nicht für den Produktivbetrieb geeignet. Openbravo basiert vollständig auf einem Webclientansatz und Adempiere verfügt derzeit nur über eine reine Clientapplikation.

Eine Plattformunabhängigkeit ist durch die Verwendung eines Javaframeworks bei Adempiere, dem vollständigen Webclientansatz bei Openbravo und der Verwendung eines Python/GTK Frameworks in TinyERP in allen drei Fällen gewährleistet. Ein Installationsaufwand für zusätzliche Clients entsteht bei Openbravo nicht. Die Clients von Adempiere und TinyERP lassen sich jedoch problemlos verteilen, wodurch der Zusatzaufwand im konkreten Anwendungsfall marginal ist.

4.5 Qualitativ-strategische Bewertung

Mit der qualitativ-strategischen Bewertung werden die in Abschnitt 4.2 definierten Ziele der Placeholder GmbH betrachtet. Zunächst ist, wie in Abschnitt 3.4 dargestellt, ein geeigneter Kriterienkatalog festzulegen. Nach erfolgter Gruppierung und Zielgruppengewichtung kann dieser Katalog mit Softwareunterstützung durch WiBe 4.0 beurteilt werden und führt im Ergebnis zur qualitativ-strategischen Bewertung der Alternativen.

Durch das Pilotprojekt wurde die Umsetzung der in Tabelle 4.2 dargestellten funktionalen Kann- und Sollkriterien mit dem Bewertungsschlüssel aus Tabelle 4.4 bewertet (siehe Tabelle 4.5). Die Musskriterien wurden nicht bewertet, da ihre vollständige Erfüllung obligatorisch ist.

| Bewertungsschlüssel | |
|---------------------|-----------------------------|
| Nutzenpunkte | Bedeutung |
| 0 | keine Verbesserung |
| 2 | sehr wenig Verbesserung |
| 4 | wenig Verbesserung |
| 6 | mittlere Verbesserung |
| 8 | deutliche Verbesserung |
| 10 | sehr deutliche Verbesserung |

Tabelle 4.4: Bewertungsschlüssel für die Kriterienerfüllung

| Erfüllung funktionaler Anforderungen | | | | | | |
|--------------------------------------|------|------|---------|-----------|-----------|-------------------|
| | Soll | Kann | TinyERP | Adempiere | Openbravo | Sage Office Line* |
| E-Mail-Funktion | X | | 8 | 5 | 8 | 5 |
| Teillieferung | X | | 5 | 10 | 8 | 5 |
| Mehrlagerfähigkeit | X | | 10 | 10 | 10 | 8 |
| Produktion / Stücklisten | X | | 10 | 10 | 10 | 8 |
| Anbindung Internetshop | | X | 10 | 7 | 3 | 0 |
| DATEV Export | | X | 0 | 6 | 0 | 10 |
| SKR03 Kontenrahmen | | X | 10 | 10 | 10 | 10 |
| Artikelgewichte | | X | 10 | 10 | 8 | 5 |
| Konsignationslager | | X | 5 | 8 | 5 | 3 |

Legende:

* Ergebnisse für Sage Office Line nach Beratungsgepräch mit Integrationspartner geschätzt.

Tabelle 4.5: Erfüllung der Kriterienliste durch die Alternativen

Neben der, im Pilotprojekt explizit untersuchten, Funktionalität (siehe Tabelle 4.5) enthält der Katalog weitere Qualitätsmerkmale (Nutzbarkeit, Leistung, Support, Zuverlässigkeit) und die strategischen Ziele (Anbieterunabhängigkeit, Weiterentwickelbarkeit, Verbesserung Workflows, Verbesserung Datenhaltung und Datenverfügbarkeit). Entsprechende Indikatoren zur Bewertung der Zielgruppen wurden festgelegt (siehe Tabelle 4.6).

| Zielgruppe | Gewicht | Untergruppe | Gewicht | Kriterien |
|--------------------|---------|---------------------------------|---------|--|
| Qualitätsziele | 75,00% | | | |
| | | Funktionalität | 30,00% | siehe Tabelle 4.5 |
| | | Nutzbarkeit | 10,00% | Webclient, Layout u. a. |
| | | Leistung | 10,00% | Geschwindigkeit, Antwortzeit u. a. |
| | | Support | 25,00% | Verfügbarkeit Dienstleister, Installation zusätzlicher Clients u. a. |
| | | Zuverlässigkeit | 25,00% | Architekturmodell, Datenhaltung u. a. |
| Strategische Ziele | 25,00% | | | |
| | | Anbieterunabhängigkeit | 5,00% | Abhängigkeit von Anbietern |
| | | Weiterentwickelbarkeit | 15,00% | Dokumentation, API's und Module u. a. |
| | | Verbesserung Workflows | 25,00% | Zeitaufwand Workflows, Verbesserung Datenqualität |
| | | Datenhaltung und -verfügbarkeit | 55,00% | Historie, Entscheiderunterstützung, Kommunikationsdienste u. a. |

Tabelle 4.6: Zielgewichtung qualitativ-strategischer Ziele

Das Erreichen der Qualitätsziele ist der Placeholder GmbH dreimal so wichtig wie die Erreichung strategischer Ziele. Die Gewichte der Untergruppen in Abstimmung mit der Placeholder GmbH stellt Tabelle 4.6 dar. Mit dem Pilotprojekt wurden die Kriterien entweder bereits analytisch bewertet (Funktionalität) oder können geschätzt werden. Nach dem Übertragen des Kriterienkataloges kann das Projekt mit der Software WiBe 4.0 bewertet werden.

Nach der Durchführung der vollständigen Bewertung ergibt sich das in Tabelle 4.7 dargestellte Ergebnis. Die CSS-Referenzalternative „Sage Office Line“ wurde zu Vergleichszwecken ebenfalls bewertet. Allerdings wurde in diesem Fall auf ein Pilotprojekt ganz verzichtet und stattdessen auf Grund eines Beratungsgesprächs mit einem Integrationspartner bewertet⁸.

| | Nutzenpunkte |
|-------------------------|--------------|
| TinyERP | 66 |
| Adempiere | 75 |
| Openbravo | 73 |
| Sage Office Line | 45 |

Tabelle 4.7: Qualitativ-strategische Alternativenbewertung

4.6 Monetäre Bewertung

Auch für die monetäre Bewertung muss zunächst ein Kriterienkatalog festgelegt werden, der alle für das Projekt relevanten Kosten- und Nutzenpositionen enthält. Darüber hinaus sind unsichere Kostenpositionen mit Wagniszuschlägen zu versehen (siehe Abschnitt 3.5). Im Anschluss erfolgt die Bewertung mit Softwareunterstützung durch WiBe 4.0.

Sowohl Server- als auch Clientinstallationen können in allen betrachteten Fällen auf vorhandener Hardware erfolgen. Ersparnisse oder Mehrkosten für Hardware sind damit nicht zu berücksichtigen. Durch das Pilotprojekt bzw. entsprechende Gespräche mit Dienstleistern konnten die Einführungs- und Betriebskosten der Alternativen bei der Placeholder GmbH monetär bewertet werden (siehe Tabellen 4.8 und 4.9). Direkte monetäre Nutzen durch die Einführung fallen nicht an. Die monetären Nutzen des Betriebes resultieren aus dem Wegfall des alten Anwendungssystems und sind in Tabelle 4.10 dargestellt. Als Betrachtungszeitraum für die Bewertung wurden fünf Jahre veranschlagt. Der Betrachtungszeitraum beginnt mit dem Jahr 2007 (Ab-

⁸ Firma Van der Steen und Partner, Husumer Strasse 8, 21465 Reinbek

schlussevaluierung), die eigentliche Umstellung ist für Anfang 2008 geplant. Abgezinst wird das Projekt auf Anfang 2007.

| Kosten der Einführung von AS-Alternativen | | | | | | |
|--|-------------------|-------------------|-------------------|-------------------------|---------------|-----------|
| Kostenart | TinyERP | Openbravo | Adempiere | Sage Office Line | Risiko | HW |
| Kosten der Abschluss-evaluierung | 896,00 € | 1500,00 € | 1500,00 € | 750,00 € | | j |
| Anpassung der Software, Customizing / Einstellung | 4480,00 € | 3500,00 € | 2500,00 € | 1500,00 € | 25,00% | j |
| Installation Anwendungs- und Datenbankserver | 896,00 € | 1792,00 € | 1792,00 € | 500,00 € | | j |
| Übername der Datenbestände | 4480,00 € | 4480,00 € | 2500,00 € | 4480,00 € | 25,00% | j |
| Einführungsschulung Anwender | 3360 € | 1500,00 € | 1000,00 € | 500,00 € | | j |
| Reisekosten Anwenderschulung | 400 € | 0,00 € | 0,00 € | 0,00 € | | j |
| Unterbringungskosten Anwenderschulung | 400 € | 0,00 € | 0,00 € | 0,00 € | | j |
| Opportunitätskosten Produktivitätsausfall Schulung | 2000 € (8 AT) | 500€ (2 AT) | 500€ (2 AT) | 500€ (2 AT) | | n |
| Unproduktives Arbeiten (Opportunitätskosten) | 3000,00 € (12 AT) | 3000,00 € (12 AT) | 3000,00 € (12 AT) | 3000,00 € (12 AT) | 25,00% | n |
| Lizenzkosten | 0,00 € | 0,00 € | 0,00 € | 11680,00 € | | j |

Legende:

HW = hausweltswirksam

AT = Arbeitstage (Placeholder GmbH)

Tabelle 4.8: Kosten der Einführung von AS-Alternativen

| Kosten des Betriebes von AS-Alternativen | | | | | | |
|---|----------------|------------------|------------------|-------------------------|---------------|-----------|
| Kostenart | TinyERP | Openbravo | Adempiere | Sage Office Line | Risiko | HW |
| Support pro Jahr (pro Jahr) | 3150,00 € | 3500,00 € | 3000,00 € | 1800,00 € | | j |
| Datensicherung, Updates, Pflege (pro Jahr) | 1792,00 € | 3000,00 € | 3000,00 € | 1792,00 € | | j |
| ungeplante Downtime (pro Jahr) | 625,00 € | 625,00 € | 625,00 € | 625,00 € | 25,00% | n |
| Lizenzen | 0,00 € | 0,00 € | 0,00 € | 1896,00 € | | j |

Legende:

HW = hauswaltswirksam

Tabelle 4.9: Kosten des Betriebes von AS-Alternativen

| Nutzen aus Betrieb | | | |
|--|-----------------------|---------------|-----------|
| Kostenart | Wegfall Primus | Risiko | HW |
| ungeplante Downtime Altsystem (pro Jahr) | 5000,00 € | 25,00% | n |
| Lizenzkostensparnis (pro Jahr) | 1800,00 € | | j |
| Datensicherung, Updates, Pflege (pro Jahr) | 1792,00 € | | j |

Legende:

HW = hauswaltswirksam

Tabelle 4.10: Nutzen aus dem Betrieb von AS-Alternativen

Aus der Durchführung der monetären Bewertung ergeben sich für die Alternativen die in Tabelle 4.11 dargestellten Risikowerte für das Bezugsjahr 2007.

| | Risikowert |
|-------------------------|-------------------|
| TinyERP | -13.548,72 € |
| Adempiere | -11.732,82 € |
| Openbravo | -17.310,26 € |
| Sage Office Line | -18.173,14 € |

Tabelle 4.11: Risikowert der AS-Alternativen

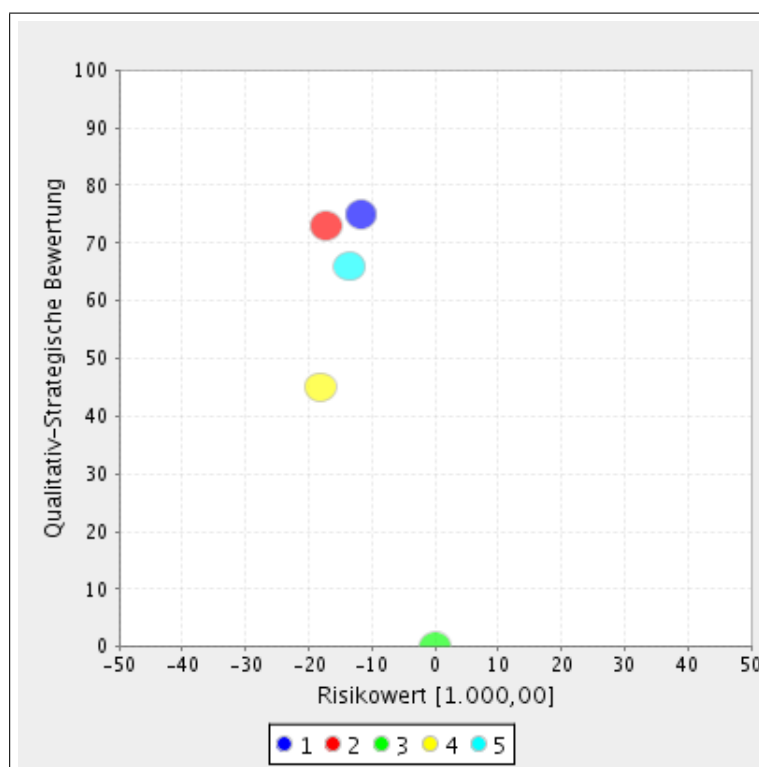
4.7 Entscheidung

Eine Entscheidung über die Durchführung des Migrationsprojektes der Placeholder GmbH, wird anhand der Ergebnisse der qualitativ-strategischen und monetären Bewertung (siehe Abschnitte 4.5, 4.6) unter Ergebnisdarstellung mit der in Abschnitt 3.6 beschriebenen Portfoliotechnik und daraus resultierenden Handlungsalternativen getroffen.

Abbildung 4.1 zeigt die Gegenüberstellung der monetären und qualitativ-strategischen Bewertung für das Migrationsprojekt der Placeholder GmbH. Alle betrachteten Alternativen, egal ob OSS oder CSS, stellen eine qualitativ-strategische Verbesserung gegenüber „Primus Handwerk“ dar.

Der Risikowert für alle AS-Alternativen ist negativ. Rein monetär betrachtet, wären somit alle Alternativen ungünstiger als die Beibehaltung des Altsystems „Primus Handwerk“, d. h. im relevanten Betrachtungszeitraum rechnet sich das Migrationsprojekt rein monetär keinesfalls.

Keine der Alternativen kann gemäß der durch Abbildung 2.6 vorgegebenen Handlungsstrategien uneingeschränkt befürwortet werden. Die strategischen Möglichkeiten der Zukunft sollten jedoch für die OSS-Alternative Adempiere kritisch geprüft werden. Diese Alternative dominiert die beiden anderen OSS-

**Legende:**

- 1 = OSS-Alternative Adempiere
- 2 = OSS-Alternative Openbravo
- 3 = Primus Handwerk
- 4 = CSS-Alternative Sage Office Line
- 5 = OSS-Alternative TinyERP

Abbildung 4.1: Beispielportfolio des Migrationsprojekts bei der Placeholder GmbH

Alternativen ebenso wie die CSS-Alternative sowohl durch einen höheren Risikowert als auch höheren qualitativ-strategischen Nutzenbeitrag.

Insbesondere mit dem steigenden Handelsvolumen auch im Ausland und im Zusammenhang mit der angestrebten stärkeren Kundenorientierung sollte geprüft werden, ob die durch Einführung der OSS-Alternative größere Flexibilität den monetären Mehraufwand rechtfertigt. Die CSS-Alternative Sage Office Line ist in jedem Fall abzulehnen, da sie weder einen monetären

Vorteil gegenüber dem Altsystem noch eine überdurchschnittliche qualitativ-strategische Verbesserung ermöglicht.

Um eine Entscheidung über die Durchführung dieses Projekts treffen zu können, müssen entsprechend Abschnitt 3.6 die Zukunftspotentiale des Einsatzes der Alternative Adempiere kritisch geprüft und gegenüber dem monetären Aufwand bewertet werden. Dies ist eine Führungsaufgabe und obliegt der Geschäftsführung der Placeholder GmbH.

4.8 Ergebniszusammenfassung und Diskussion des Modells

Die Placeholder GmbH steht seit langem vor der Entscheidung über die Einführung eines neuen Anwendungssystems zur optimalen Unterstützung ihrer betrieblichen Aufgaben. Die Geschäftsleitung war jedoch nicht im Stande eine derart weitreichende Entscheidung alleine zu treffen. Erstens fehlte ihr hierzu die Kenntnis über vorhandene Alternativen und zweitens Methoden, eine derart weitreichende Entscheidung fundiert treffen zu können. Durch Anwendung des in Kapitel 3 erarbeiteten Entscheidungsmodells wurde zunächst eine Projektvorstudie durchgeführt.

Mit der Projektvorstudie wurden die qualitativen und strategischen Ziele der Einführung eines neuen Anwendungssystems für die Placeholder GmbH festgelegt. Anhand der Zielstruktur ist zu erkennen, dass die Placeholder GmbH die effiziente Unterstützung der betrieblichen Abläufe und die Ertragssteigerungen durch Verbesserung der Kundenkommunikation stark gewichtet. Innovationsstrategische Überlegungen in Verbindung mit dem IT-Einsatz sind für die Placeholder GmbH dagegen von nachrangiger Bedeutung. Wie die Projektvorstudie weiter zeigen konnte, sind wichtige Vorteile freier Software wie die Veränderbarkeit und die Anbieterunabhängigkeit für die Placeholder GmbH von nachrangiger Bedeutung. Die mit der Projektvorstudie

durchgeführte Risikobestimmung des Projektes zeigt, dass die Risiken der Einführung eines ERP-Systems zwar insgesamt hoch, aber bei der vorliegenden Unternehmensgröße durchaus handhabbar sind.

Der Projektvorstudienbericht fasst zu diesem Zweck die Ergebnisse der Projektabgrenzung, Zieldefinition, Risikoanalyse und Ressourcenermittlung zusammen. Durch die IST-Analyse (Hardware, Software, Prozesse, Personal) konnte die weitreichende Bedeutung des Projekts für die Placeholder GmbH bestätigt werden. Die Aufnahme der IST-Situation war dabei wichtige Eingangsvoraussetzung der technischen Machbarkeitsanalyse. So konnte nach Ermittlung und Auswahl einer geeigneten Alternative die technische Durchführbarkeit des Migrationsprojektes durch Pilotinstallationen bestätigt werden.

Insbesondere die Durchführbarkeit des Migrationsprojekts auf der vorhandenen Hardware konnte damit geprüft und für alle selektierten Alternativen bestätigt werden. Aus der technischen Machbarkeitsstudie sind wichtige Kosten-/Nutzenpositionen für die monetäre Bewertung abgeleitet worden. Die technische Analyse der gewählten Alternativen bildete darüber hinaus die Grundlage zur qualitativ-strategischen Bewertung. Mit der Erstellung der projektspezifischen Kriterienkataloge für die monetäre und die qualitativ-strategische Betrachtung konnten die entsprechenden Bewertungen durchgeführt und mittels der Portfoliotechnik anschaulich dargestellt werden. Bei diesem Anwendungsbeispiel wurde auf die explizite qualitativ-strategische Betrachtung von Dringlichkeitsfaktoren und externer Wirkung des Projekts durch die Erhebungsbereiche WiBe D und WiBe E verzichtet. Relevante Dringlichkeitsfaktoren (Stabilität des Altsystems, Flexibilität) wurden stattdessen im Kriterienkatalog der WiBe Q miteingefasst. Externe wirtschaftliche Folgeeffekte des Projekts (WiBe E) sind für die Entscheidung der Placeholder GmbH nicht von Belang.

Durch die Projektvorstudie und die IST-Analyse wurde die Tragweite der Entscheidung für die Placeholder GmbH erstmalig zusammenfassend darge-

legt. Die Alternativenermittlung und -prüfung ermöglichte die auf der monetären und qualitativ-strategischen Bewertung beruhende Wirtschaftlichkeitsbetrachtung des Migrationsprojektes. Es konnte gezeigt werden, dass die rein monetäre Betrachtung sofort zur Ablehnung aller Alternativen führen würde⁹. Die Bewertung der qualitativen und strategischen Merkmale der Alternativen ergibt jedoch, dass zumindest durch die OSS-Alternativen ein signifikanter Beitrag zur Erreichung dieser Ziele geleistet werden kann. Die OSS-Alternative Adempiere dominiert dabei alle anderen Alternativen. Die Geschäftsleitung der Placeholder GmbH ist damit in der Lage zu entscheiden, ob ihr dieser Beitrag der Einführung der OSS-Alternative Adempiere 11.732,82 € wert ist (siehe Tabelle 4.11).

Die Anwendung des Entscheidungsmodells hat darüber hinaus gezeigt, dass z. B. eine Aussage über die monetären Konsequenzen von Systemausfällen wesentlichen Einfluss auf die gesamte monetäre Bewertung haben kann. Dieser Wert ist ex post über die Opportunitätskosten und Beseitigungskosten bestimmbar. Eine Aussage über Systemausfälle der im Einsatz befindlichen Lösung ist für die Zukunft allerdings ebensowenig möglich wie für die betrachteten Alternativen. Je nach relativem Kosten- bzw. quantitativem Nutzenanteil¹⁰ und Bewertung mit einem entsprechendem Risikofaktor kann die monetäre Aussage damit instabil, d. h. ungenau, werden.

Auch für die Beurteilung von qualitativen und strategischen Merkmalen ist ein Pilotprojekt nur eine Hilfe. Viele Einschätzungen beruhen auf der Güte der Indikatoren und der Expertenmeinung der Entscheidungsvorbereiter. Die Berücksichtigung entsprechender Risikomultiplikatoren auch bei der qualitativ-strategischen Bewertung fehlt dem hier vorgestellten Entscheidungsmodell noch.

Die beispielhafte Anwendung des Entscheidungsmodells konnte zeigen, dass besondere qualitative und/oder strategische Merkmale von OSS entschei-

⁹ da alle Alternativen einen negativen Risikowert haben

¹⁰ durch Wegfall des instabilen Altsystems

dungsrelevant sein können (z. B. Anbieterunabhängigkeit, Interoperabilität, Stabilität), aber nicht sein müssen. Im vorgestellten Fall sind die erweiterten Nutzungsrechte, wie das Recht zur Veränderung des Quellcodes oder dessen Verbreitung, nicht entscheidungsrelevant. Durch die qualitativ-strategische Bewertung werden AS-Alternativen nach den aus der Unternehmens- bzw. IT-Strategie abgeleiteten Zielen bewertet. Besonderheiten von OSS, wie etwa spezielle Nutzungslizenzen, Geschäftsmodelle u. a., können durch Verwendung der Nutzwertanalyse im WiBe-Verfahren vollständig erfasst und bewertet werden. Die Bewertung der Alternativen resultiert damit aus den unternehmens- und fallspezifischen Zielen. Das Vorgehen für OSS-Alternativen kann wie im Anwendungsfall dargestellt auch für CSS-Alternativen angewendet werden.

Kapitel 5

Zusammenfassung und Ausblick

Kapitel 1 enthält eine allgemeine Einführung in das Thema Open Source Software und beschreibt die Entwicklung der Softwareindustrie. Es wird dargestellt wie Open Source Software in der Wirtschaft aktuell eingesetzt wird und welche Besonderheiten das Open Source Entwicklungsparadigma auszeichnen. Anhand der aktuellen Situation werden die zu lösenden Probleme und die Konsequenzen für den zukünftigen Wettbewerb in der Softwareindustrie erläutert. Besonders wird dabei auch auf die Haltung der Staaten und deren Wirkung eingegangen. Um ein fundiertes Urteil über den Einsatz von OSS treffen zu können, muss der Entscheider die dargestellte aktuelle Ausgangssituation und die Einflussfaktoren für die zukünftige Entwicklung kennen.

In Kapitel 2 werden die theoretischen Grundlagen betrachtet, die bei einer zu treffenden Einsatzentscheidung unbedingt bekannt sein sollten. Dabei wird insbesondere auf die elementare Bedeutung der Qualität, sowohl auf Ebene der Entscheidungsfindung als auch im Rahmen der produktbezogenen Bewertung von Anwendungssystemen, eingegangen. Es wird gezeigt, dass beim Einsatz eines neuen Anwendungssystems die Entscheidung immer strategische, technische und ökonomische Faktoren berücksichtigen sollte. Während die vorgestellten technischen Grundlagen und ökonomischen Bewertungsmethoden für OSS und CSS i. d. R. gleichermaßen gelten, stellen insbesondere die Rechtsgrundlagen und die aus dem offenen Entwicklungsprozess resultierenden strategischen Vorteile von OSS besondere Faktoren

beim Einsatz dieser Softwaregattung dar. Eine klassische kostenbasierte Bewertung von Anwendungssystemen genügt demnach insbesondere dann nicht mehr, wenn langfristige strategische Faktoren berücksichtigt werden müssen. Danach werden Methoden zur Kostenbewertung und Methoden zur Bewertung von qualitativ-strategischen Nutzen im Einzelnen vorgestellt. Es wird gezeigt, dass das mehrdimensionale WiBe-Verfahren der KBSt, obwohl für die öffentliche Verwaltung konzipiert, ausreichende Flexibilität aufweist, um – auf die Bedürfnisse kommerzieller Unternehmen angepasst – als Bewertungsverfahren im neu entwickelten Modell zu dienen. Dadurch lässt sich der Bewertungsrahmen hin zu einer umfassenden Sichtweise auf Migrations-/Substitutionsprojekte erweitern.

Im Kapitel 3 wird das Entscheidungsmodell (siehe Abbildung 5.1) vorgestellt. Anders als alle anderen vorgestellten Verfahren erfordert das neue Modell eine Projektvorstudie und konkrete Entscheidungsregeln, die über eine rein monetäre Betrachtung hinaus die qualitativ-strategischen Anforderungen der Unternehmen berücksichtigen und damit einen wichtigen Beitrag zu deren Behauptung im Wettbewerb leisten können.

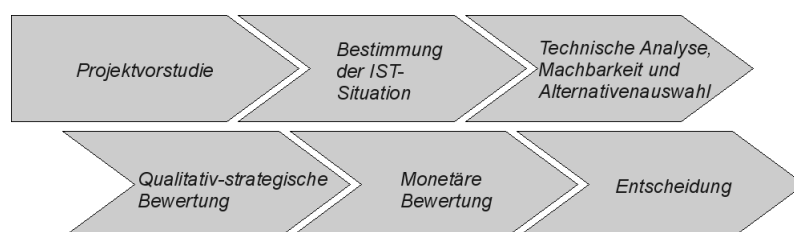


Abbildung 5.1: Entscheidungsmodell

Die Projektvorstudie erhöht darüber hinaus die Durchführungsqualität bei der Entscheidungsvorbereitung. Sie stellt insbesondere sicher, dass nur Projekte einer weiteren Bewertung unterzogen werden, die technisch wie ökonomisch prinzipiell machbar sind und keine nicht handhabbaren Risiken aufweisen. Das für OSS besondere Vorgehen zur Alternativenauswahl wird vorgestellt. Mit der Prüfung der technischen Machbarkeit durch ein geeignetes Pilotprojekt und der zuvor bestimmten IST-Situation kann die

qualitativ-strategische und monetäre Bewertung durchgeführt und damit eine Entscheidung unter Berücksichtigung aller relevanten Faktoren getroffen werden. Da sich die quantitative (monetäre) und qualitativ-strategische Bewertung nicht im gleichen Maßstab gegenüberstellen lassen, wurde die Portfoliotechnik zur Ergebnisdarstellung verwendet. Die Portfoliotechnik erlaubt es dem Führungspersonal, entweder eine direkte Entscheidung zu treffen oder in Wahrnehmung seiner Führungsaufgabe gezielt Einfluss auf die Projektkonfiguration zu nehmen.

In Kapitel 4 der Arbeit wurde mit dem Fallbeispiel der Placeholder GmbH die praktische Anwendung des entwickelten Entscheidungsmodells demonstriert. Dabei konnte gezeigt werden, dass mit Hilfe des Verfahrens AS-Alternativen nicht nur ermittelt, sondern auch bewertet werden können. Auf Grund der Tatsache, dass sowohl quantitative als auch der qualitative Faktoren der Alternativen bewertet werden, kann ein differenzierteres Urteil über die Vorteilhaftigkeit der Alternativen getroffen werden als das mit Hilfe einer reinen Kostenanalyse möglich wäre.

Es konnte gezeigt werden, dass, obwohl beide Alternativen über die veranschlagte Nutzungsdauer von fünf Jahren einen negativen Kapitalwert gegenüber der gegenwärtig genutzten Branchenlösung haben, nicht quantifizierbare Nutzen der OSS-Alternativen einen wesentlichen Beitrag zur Erreichung der von der Placeholder GmbH definierten Ziele leisten können. Damit ist das Management in der Lage zu entscheiden, ob die bessere Erreichung der Ziele den finanziellen Mehraufwand rechtfertigt.

Die Anwendung des Modells hat darüber hinaus gezeigt, dass hohe Konzentration einzelner Kosten- und/oder Nutzenpositionen und deren teilweise schwierige Vorhersagbarkeit zu einer Instabilität der quantitativen Betrachtung führen. Ebenso setzen die Gruppierung und Gewichtung von qualitativen und strategischen Zielen ein Werturteil voraus. Die Gewichtung hat damit erhebliche Auswirkung auf das Ergebnis. Sie beruht auf dem Know-how

der Entscheidungsvorbereiter und Dienstleister ebenso wie auf der Kenntnis der spezifischen Unternehmensziele.

Zukünftige wissenschaftliche Bemühungen müssen das in dieser Arbeit vorgestellte Entscheidungsmodell durch empirische Analyse validieren. Aus dem so entstehenden Erfahrungsschatz bei der Anwendung des Modells sollten Vorlagen für die Vielfalt denkbarer Anwendungsszenarien erstellt werden. Zum einen sollte dieses Vorgehen bei der Benennung der fallspezifischen Ziele helfen, zum anderen eine Kriterienvorlage zur quantitativen und qualitativen Bewertung darstellen.

Wie in der Einleitung der Arbeit (siehe Kapitel 1) festgestellt wurde, ist eine wissenschaftlich fundierte Prognose über die Zukunft des Wettbewerbs in der Softwareindustrie nur schwierig zu treffen. Sicher ist jedoch, dass OSS längst dem Status einer „technischen Spielerei“ entwachsen ist und eine reale Konkurrenz für CSS in vielen Anwendungsbereichen darstellt (vgl. Méndez et al., 2005; Ghosh, 2006, , S. 19 ff.).

In Kapitel 1 wurde bereits erwähnt, dass die rechtliche Entwicklung erheblichen Einfluss auf den Wettbewerb nehmen kann. Zumindest der Prozess zwischen SCO und IBM über Patentverletzung durch zentrale Linuxkomponenten scheint aktuell zu Gunsten von IBM entschieden zu sein (vgl. Laube, 2007). Darüber hinaus trat am 1. Juli die neue GPL v3 in Kraft, die u. a. Regelungen zum Digital Rights Management (DRM) und Softwarepatenten beinhaltet. Sie ist wesentlich stärker als die GPL v2 auf internationale Wirksamkeit ausgelegt (vgl. Schneider, 2007). Es bleibt abzuwarten, ob die GPL v3 von der OSS-Bewegung auf breiter Front akzeptiert wird und wie sich die Bemühungen um Rechtssicherheit in der internationalen Rechtspraxis auswirken.

Für das Entscheidungsmodell hat die neue GPL v3 kaum direkte Konsequenzen. Indirekt stärkt sie jedoch die Rechtssicherheit beim Einsatz von unter ihr lizenzierter OSS. So werden dem Nutzer erstmals explizit die vollen Nutzungsrechte verwendeter Softwarepatente eingeräumt. Die Regelungen zum

DRM erlauben zwar explizit die Umsetzung von digitalem Rechtemanagement durch OSS, stellen aber gleichzeitig sicher, dass zu Grunde liegende Verfahren ebenfalls quelloffen sind und daher umgangen werden dürfen.

Zusammenfassend kann gesagt werden, dass zumindest die rechtliche Entwicklung aktuell auf Sicherheit zu Gunsten der Nutzer abzielt. Die technischen Vorteile durch den offenen Entwicklungsprozess sprechen ebenfalls häufig für den Einsatz von OSS. Beim Einsatz eines neuen Anwendungssystems musste in der Vergangenheit eine Prognose über die Zukunft eines speziellen Marktteilnehmers und dessen Lizenzpolitik getroffen werden. Mit Verbreitung des OSS Paradigmas liegt IT-Strategien nunmehr eine Prognose über den Wettbewerb des besten Konzepts im Gesamtmarkt zu Grunde.

Kapitel A

Anhang zu Theoretische Grundlagen

A.1 Antitrust Gesetz in den USA

Das Anti-Trust Gesetz in den Vereinigten Staaten von Amerika wurde 1890, unter anderem zum Schutz vor dem Missbrauch von wirtschaftlichen Monopolen (vgl. Sherman, 2007), entwickelt. Danach werden Unternehmen, welche z. B. unerlaubt Preisabsprachen treffen, als kriminell angesehen und entsprechend rechtlich belangt. Die Zerschlagung des Standard Oil Imperiums von John D. Rockefeller im Jahre 1911 war der erste Anwendungsfall dieses Gesetzes. Auch IBM wurde im Jahr 1969, basierend auf diesem Gesetz, Missbrauch des Marktmonopols vorgeworfen. IBM entschloss sich daraufhin Software und Hardware künftig voneinander zu trennen. Die Anklage wurde schließlich 1982 fallen gelassen (vgl. Stielor, 2002).

A.2 Magisches Dreieck des Projektmanagement

Das Magische Dreieck ist eine Form der Veranschaulichung für die Projektzieldimensionen Zeit, Kosten, Leistung / Qualität (siehe Abbildung A.1).

Die Kanten des Dreiecks machen dabei die Abhängigkeiten der einzelnen Dimensionen untereinander deutlich. Höhere Produktqualität bedeutet zum Beispiel häufig größeren Faktoreinsatz und somit höhere Kosten (vgl. Gaulke, 2004, S. 5).

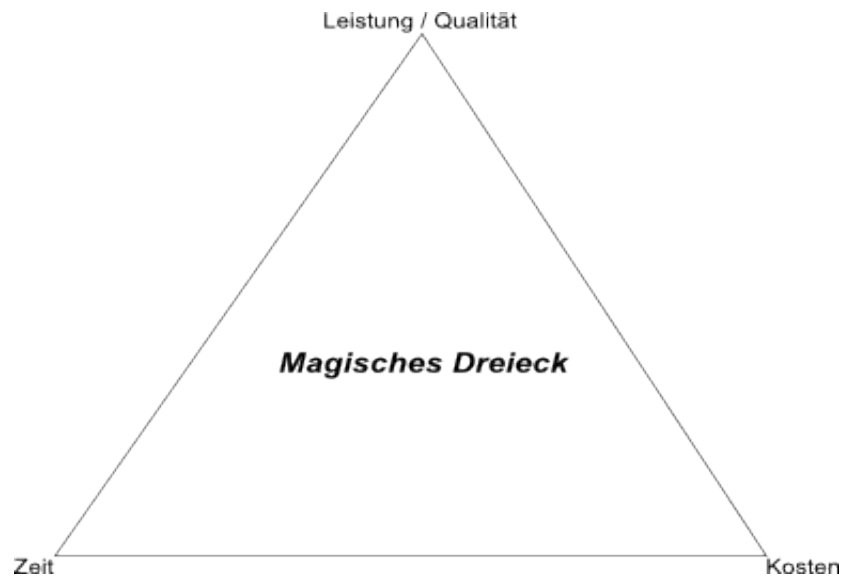


Abbildung A.1: Magisches Dreieck nach Gaulke (vgl. Gaulke, 2004, S.6)

Kapitel B

Modellhilfsmittel

B.1 Projektrisiko-Checkliste

| Projektrisikobereich | Risikofaktor | Bewertung |
|---------------------------|---|-----------|
| Geschäftliche Ausrichtung | Kritikalität des IT-Systems | 4 |
| | Fehlende Unterstützung durch das Management | 2 |
| | Instabilität der Organisation | 1 |
| | Dynamik des Marktumfeldes | 2 |
| | Umfang rechtlicher Vorgaben | 2 |
| | Abweichung von der IT-Strategie | 1 |
| Projektmanagement | Mangel an eigenem Know-how | 5 |
| | Verhältnis fremdes eigenes Know-how | 5 |
| | Anzahl extern / intern zu beschaffender Projektbeteiligter | 2 |
| | Verhältnis externer / interner Projektbeteiligter | 2 |
| | Budgetmangel für derartige Projekte | 2 |
| | Anzahl involvierter Unternehmensstandorte / Unternehmensteile | 2 |
| | Abhängigkeit von anderen Projekten | 1 |
| Geschäftsprozesse | Fehlendes Verständnis vorhandener Geschäftsprozesse | 3 |
| | Auswirkung auf zukünftige Geschäftsprozesse | 3 |
| Anwender | Mangel an Organisationsflexibilität | 3 |
| | Mangel an Zielkorrelation Mitarbeiter / Führungskräfte | 4 |
| Technologie | Komplexität des Anwendungssystems | 3 |
| | Komplexität der Integration in das Anwendungssystemumfeld | 2 |
| | relativer Anteil beteiligter Server- und Hostsysteme | 4 |
| | relativer Anteil beteiligter Clientsysteme | 4 |
| | relativer Anteil beteiligter Systemperipherie | 3 |
| Daten | Fehlende Informationsarchitektur | 4 |
| | Unzureichend dokumentierte Datenspeicher | 3 |
| | Mangelnde Datenqualität | 2 |
| | Anzahl der Datenspeicher | 1 |

Tabelle B.1: Projektrisikocheckliste (vgl. Gaulke, 2004, S. 74 ff.)

B.2 Bewertung des Risikos und Empfehlung

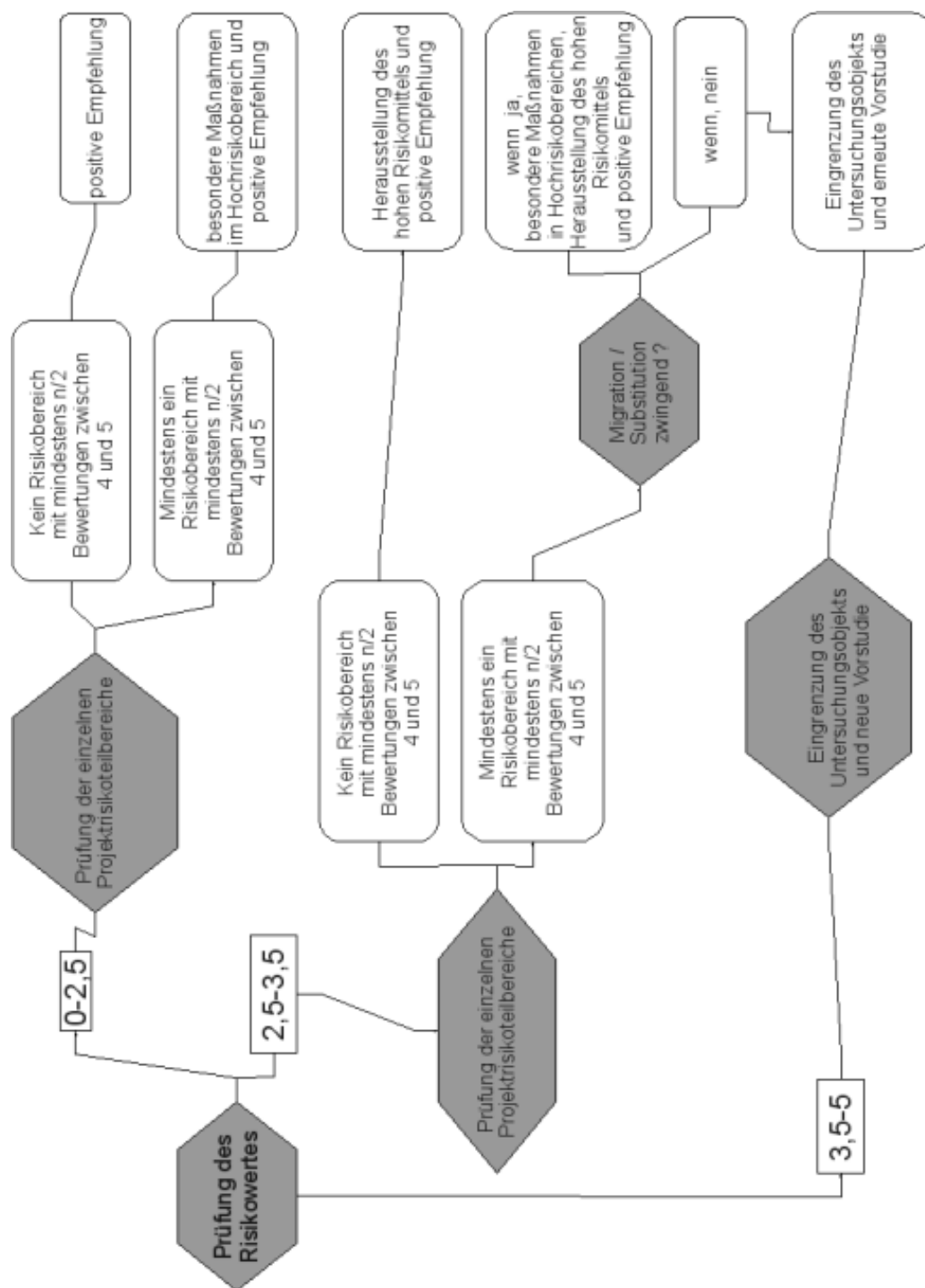


Abbildung B.1: Bewertung des Risikos und Empfehlung

B.3 Erhebungshilfen

| Hardware | |
|-----------------|---|
| Server | |
| Inventar-Nr. | |
| F-HW-S-0001 | |
| Architektur | i386 |
| Prozessoren | Pentium IV |
| Taktfrequenz | 2,8Ghz |
| Arbeitsspeicher | 1,0 GB |
| Festspeicher | Raid 5 / 400GB |
| Netzwerk | 100MBit |
| Betriebssystem | W2003SBS |
| Standort | HS |
| Abteilung | Zentrale |
| Administrator | CS24 |
| Kosten pro Jahr | 500.- € |
| Verweise | F-HW- SO- 0001, F-HW- NW- 0001, F-HW- NW- 0002, F-HW- NW- 0003 |
| Anmerkung | |

Tabelle B.2: Server Erhebungshilfe

| Hardware | | | | | | | | | | | | | | | | | | |
|--------------|-------------|-------------|-----------|--------------|-----------------|--------------|---------------------------|------------------|----------|-----------|-------------|---------------|-------------------------|-----------------------|------------------------------|-----------------|---------------------------|-----------|
| Clients | | | | | | | | | | | | | | | | | | |
| Inventar-Nr. | Klasse | Architektur | Prozessor | Taktfrequenz | Arbeitsspeicher | Festspeicher | Netzwerk | Betriebssystem | Standort | Abteilung | Hauptnutzer | Administrator | zentrale Administration | zentrale Installation | Sonderhardware | Kosten pro Jahr | Verweise | Anmerkung |
| F-HW-C-0001 | Full-client | EM64T | Pentium-D | 3,0Ghz | 1,0 GB | 250GB | RTL 8139 | Windows XP Home | HS | Z | GF | CS24 | J | N | DVD-DL-RW, Agere V. 92 Modem | 250,- € | F-HW-S-0001, F-HW-NW-0003 | |
| F-HW-C-0002 | Full-client | i386 | Athlon-XP | 1,0 Ghz | 256 MB | 40GB | Via Rhine | Windows XP Prof. | HS | Z | RS | CS24 | J | J | N | 250,- € | F-HW-S-0001, F-HW-NW-0003 | |
| F-HW-C-0003 | Full-client | i386 | Athlon-XP | 2,1 Ghz | 512 MB | 20GB | SiS735 | Windows 2000 | HS | Z | DW | CS24 | N | N | N | 250,- € | F-HW-S-0001, F-HW-NW-0003 | |
| F-HW-C-0004 | Notebook | i386 | Pentium-M | 1,4 Ghz | 512 MB | 60GB | RTL 8139 / IN-TEL BC 2200 | Windows XP Home | HS | Z | DW | CS24 | N | N | Agere V. 92 Modem | 250,- € | F-HW-S-0001, F-HW-NW-0004 | |

Tabelle B.3: Clients Erhebungshilfe

| Hardware | | | | | | | | | | | | | | | | |
|--------------|------------|--------|------------|----------|---------|------------|----------|-----|--------------|-------|----------|--------------------|---------|----------|---------------|---|
| Netzwerk | | | | | | | | | | | | | | | | |
| Inventar-Nr. | Komponente | | | Standard | | | | | | | | Anmerkung | | | | |
| | Router | Switch | Steckkarte | Ethernet | Myrinet | Infiniband | Quadrics | ATM | Fibrechannel | iSCSI | Sonstige | | Produkt | Standort | Administrator | Kosten pro Jahr |
| F-HW-NW-0001 | | | X | X | | | | | | | | 3Com 3C940 | HS | CS24 | keine | F-HW-S-0001 |
| F-HW-NW-0002 | | | X | | | | | | | | X | AVM Fritzcard PCI | HS | CS24 | keine | F-HW-S-0001 |
| F-HW-NW-0003 | | | | X | | | | | | | | Level One 8-Port | HS | CS24 | keine | F-HW-S-0001, F-HW-C-000(1-3), F-HW-D-0001 |
| F-HW-NW-0004 | X | | | X | | | | | | | | Linksys WRT54G v.2 | HS | CS24 | keine | F-HW-C-0004 |

Tabelle B.4: Netzwerk Erhebungshilfe

| Hardware | | | | | | | | | | | | | | | | | | | | |
|-----------------|--------------|-------|--------------|------------|-----------|----------|-----------|-------------|---------------|-----------------|-------------|-----------|--|--|--|--|--|--|--|--|
| Druck | | | | | | | | | | | | | | | | | | | | |
| | Inventar-Nr. | Alter | Drucktechnik | Hersteller | Modell | Standort | Abteilung | Hauptnutzer | Administrator | Kosten pro Jahr | Verweise | Anmerkung | | | | | | | | |
| Netzwerkdrucker | F-HW-D-0001 | | Color-Laser | Kyocera | FS-C5016N | HS | Zentrale | Alle | Placeholder | 150.- € | F-HW-S-0001 | | | | | | | | | |
| Netzwerkdrucker | F-HW-D-0002 | | SW-Laser | Brother | HL-1850 | HS | Zentrale | Alle | Placeholder | 100.- € | F-HW-S-0001 | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | |

Tabelle B.5: Druck Erhebungshilfe

| Hardware | | | | | | | | | | | | |
|-----------------|-----------|----------|------------|------------------|----------|-----------|---------------|-----------------|-------------|-----------|--|--|
| Sonstige | | | | | | | | | | | | |
| Inventar-Nr. | Alter | Embedded | Hersteller | Modell | Standort | Abteilung | Administrator | Kosten pro Jahr | Verweise | Anmerkung | | |
| USV | 1,5 Jahre | N | Riello | Win-Dialog Plus | HS | Zentrale | Placeholder | 50.- € | F-HW-S-0001 | | | |
| Telefonanlage | 5 Jahre | J | DTAG | T-Concept XI 721 | HS | Zentrale | Placeholder | keine | | | | |
| Systemtelefon | 5 Jahre | N | DTAG | T-Concept PX 722 | HS | Zentrale | Placeholder | keine | F-HW-C-0001 | | | |
| Systemtelefon | 5 Jahre | N | DTAG | T-Concept PX 722 | HS | Zentrale | Placeholder | keine | F-HW-C-0002 | | | |
| Systemtelefon | 5 Jahre | N | DTAG | T-Concept PX 722 | HS | Zentrale | Placeholder | keine | F-HW-C-0003 | | | |

Tabelle B.6: Sonstige Erhebungshilfe

| Software | | | | | | | | | | |
|-------------------------------------|--------|------------|----------------------|---------|-------------|----------|---------------|-----------------|-----------------|--|
| Betriebssysteme | | | | | | | | | | |
| Name | Anzahl | Hersteller | Version | Support | Architektur | Realtime | Administrator | Kosten pro Jahr | Verweise | Anmerkungen |
| Windows XP | 2 | Microsoft | Home Edition | N | i386 | N | CS24 | 150,00 | F-HW-C-000(1,4) | Kosten gelten pro Lizenz |
| Windows XP | 1 | Microsoft | Professional Edition | N | i386 | N | CS24 | 100,00 | F-HW-C-0002 | |
| Windows 2000 | 1 | Microsoft | SP4 | N | i386 | N | CS24 | 100,00 | F-HW-C-0003 | |
| Windows 2003 Small Business Edition | 1 | Microsoft | R2 | J | i386 | N | CS24 | 350,00 | F-HW-S-0001 | Serverbetriebssystem erfordert mehr Adminaufwand |

Tabelle B.7: Betriebssysteme Erhebungshilfe

| Software | | | | | | | | | | |
|---|----------------|-------------------|---------|---------------|---------|-----------|---------------|-----------------|-------------|-------------|
| Serverdienste | | | | | | | | | | |
| Bereich | Betriebssystem | Produkt | Version | Hersteller | Support | Abteilung | Administrator | Kosten pro Jahr | Verweise | Anmerkungen |
| Infrastrukturdienste - Riello UPS Überwachung - Windows 2003 Faxserver | W2003SBS | UPSMON Monitoring | 5.0.5 | Riello | N | Zentrale | CS24 | 50,00 | F-HW-S-0001 | |
| | W2003SBS | W2003SBS | R2 | Microsoft | J | Zentrale | CS24 | 100,00 | F-HW-S-0001 | |
| System- und Managementdienste | | | | | | | | | | |
| VPN- und Fernzugriff | | | | | | | | | | |
| Verzeichnisdienste | | | | | | | | | | |
| Mail- und Webdienste | | | | | | | | | | |
| Druck- und Filedienste - Windows 2003 Fileserver - Windows 2003 Druckserver | W2003SBS | W2003SBS | R2 | Microsoft | J | Zentrale | CS24 | 150,00 | F-HW-S-0001 | |
| | W2003SBS | W2003SBS | R2 | Microsoft | J | Zentrale | CS24 | 100,00 | F-HW-S-0001 | |
| Multimediendienste | | | | | | | | | | |
| Sicherheitsdienste | | | | | | | | | | |
| Datenbankdienste - Amber CS++ | W2003SBS | CS++ | 6.20 | AmberSoftware | N | Zentrale | CS24 | 100,00 | F-HW-S-0001 | |

Tabelle B.8: Serverdienste Erhebungshilfe

| Software | | | | | | | | | |
|------------------|--------|------------------|---------|---------|---------------|-----------------|-----------------|-------------|--|
| Standardsoftware | | | | | | | | | |
| Name | Anzahl | Hersteller | Version | Support | Administrator | Kosten pro Jahr | Verweise | Anmerkungen | |
| Acrobat Reader | 5 | Adobe | 8.0 | N | CS24 | | F-HW-C-000(1-4) | | |
| PDF-Creator | 4 | OpenSource (GPL) | 0.93 | N | CS24 | | F-HW-C-000(1-4) | | |
| Mozilla Firefox | 5 | OpenSource (GPL) | 2.0 | N | CS24 | | F-HW-C-000(1-4) | | |
| MS Office | 4 | Microsoft | 2003 | N | CS24 | | F-HW-C-000(1-4) | | |
| Primus Handwerk | 2 | Sage | 6.23 | J | CS24 | 900,00 | F-HW-C-000(1-4) | | |

Tabelle B.9: Standardsoftware Erhebungshilfe

| Software | | | |
|--------------------|--------|-----------------|---------|
| Individualsoftware | | | |
| Name | Anzahl | Hersteller | Version |
| | | Support | |
| | | Administrator | |
| | | Kosten pro Jahr | |
| | | Verweise | |
| | | Anmerkungen | |

Tabelle B.10: Individualsoftware Erhebungshilfe

| Prozesse | | | | | | |
|--------------------------------------|-------------|----------|--------------------------------------|--------------------------|---|--|
| Dokumentenvorlagen und Makros | | | | | | |
| | | | Komplexität Dokumentenvorlage | Komplexität Makro | | |
| Name | Speicherort | | < 0,5 MT | < 0,5 MT | | |
| | | Anzahl | 0,5 - 2 MT | 0,5 - 2 MT | | |
| | | Summe MT | 2 - 4 MT | 2 - 4 MT | | |
| | | | > 4 MT | > 4 MT | | |
| Vorlagen Versand | F-HW-S-0001 | 2 | | | X | |

Legende:

- < 0,5 MT = geringe Komplexität
- 0,5-2 MT = mittlere Komplexität
- 2-4 MT = hohe Komplexität
- > 4 MT = sehr hohe Komplexität

Tabelle B.11: Dokumentenvorlagen und Makros

| Prozesse | | | | | | | | | | | | |
|---------------------------------|------------|-------|------|-----------|-------|------|------------|-------|------|-----------|-------|------|
| Workflows | | | | | | | | | | | | |
| Client- / Serverbetriebssysteme | | | | | | | | | | | | |
| Client-Betriebssystem(e) | | | | | | | | | | | | |
| Name | eingesetzt | | | verfügbar | | | eingesetzt | | | verfügbar | | |
| | Windows | Linux | Unix | Windows | Linux | Unix | Windows | Linux | Unix | Windows | Linux | Unix |
| Primus Handwerk | X | | | X | | | X | | | X | | |

Tabelle B.13: Workflows - Client- / Serverbetriebssysteme

| Prozesse | | | | | | | | | | |
|--------------------------------------|------------------|-----------|------------|--------------------|------------|-----------|------------|-----------|------------|-----------|
| Workflows | | | | | | | | | | |
| Datenbanksysteme, Applikationsserver | | | | | | | | | | |
| Name | Datenbanksysteme | | | Applikationsserver | | | verfügbar | | | |
| | eingesetzt | verfügbar | eingesetzt | verfügbar | eingesetzt | verfügbar | eingesetzt | verfügbar | eingesetzt | verfügbar |
| Primus Handwerk | X | | | | X | | | | | |
| | Windows | Linux | Unix | Windows | Linux | Unix | Windows | Linux | Unix | |

Tabelle B.14: Workflows - Datenbanksysteme, Applikationsserver

| Prozesse | | | | | | | | | | | | |
|--|------------------------|------------------------|-----------|------------------------|------------------------|-----------|--------------|------------------|--------------------|-------------------|-------|-------------|
| Workflows | | | | | | | | | | | | |
| Benutzer-/ Rechteverwaltung und Schnittstellen | | | | | | | | | | | | |
| | | Verwaltung | | Schnittstellen | | | | | | | | |
| Name | Benutzer | | | Rechte | | | | | | | | |
| | Innerhalb d. Anwendung | über Verzeichnisdienst | Sonstiges | Innerhalb d. Anwendung | über Verzeichnisdienst | Sonstiges | IT-Verfahren | Standardsoftware | Individualsoftware | Dokumentenvorlage | Makro | Komplexität |
| Primus Handwerk | X | | | X | | | | X | | X | | gering |

Tabelle B.15: Workflows - Benutzer- / Rechteverwaltung und Schnittstellen

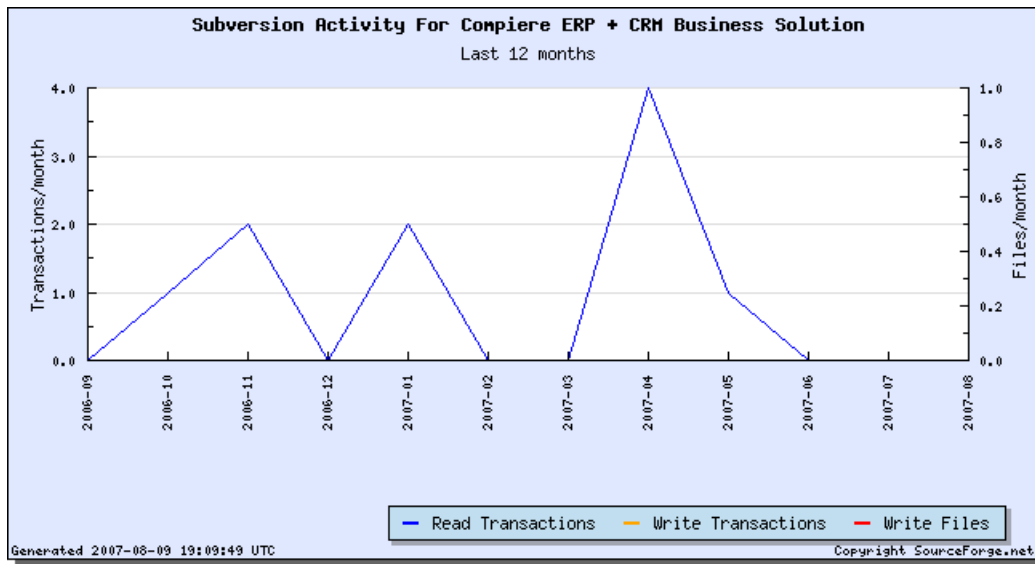
| Prozesse | | | | | | | | | | |
|--|------------------|---------|-------------|-----------|-----------------|----------------------------|-------------|---------------|-------------|--|
| Workflows | | | | | | | | | | |
| Verantwortung, Entwicklung, Hosting, Ausprägung | | | | | | | | | | |
| Name | Verantwortlicher | Hosting | Entwicklung | | | Ausprägung / Anforderungen | | | | |
| | | | MT intern | MT extern | Entwicklungs-U. | Support durch | Komplexität | Verfügbarkeit | Support | |
| Primus Handwerk | CS24, SAGE | inhouse | | | | | hoch | hoch | mittel/hoch | |

Tabelle B.16: Workflows - Verantwortung

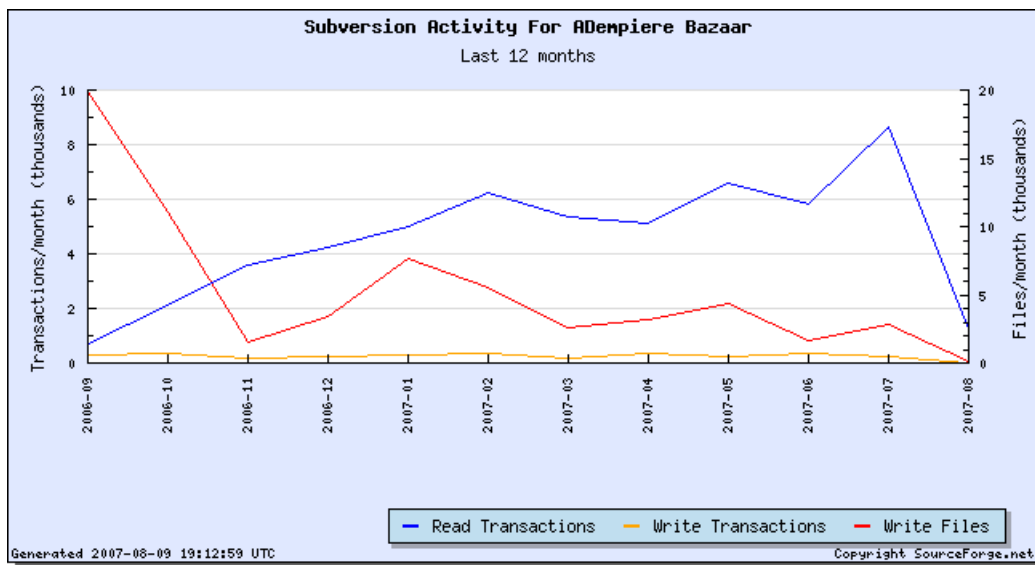
| Prozesse | | | | | | | | | | | | |
|-------------------------------|----------------|----------|-------------|--------------------|---------------------|---------------------|---------------------|-------------------|-------------|---------------------|----------|-----------|
| Workflows | | | | | | | | | | | | |
| Ausblick, Kosten, Anmerkungen | | | | | | | | | | | | |
| Name | Ausblick | | | | Kosten | | | | Anmerkungen | | | |
| | Fortschreibung | Ablösung | Einstellung | verfügbar u. Linux | Plattformunabhängig | Support intern Jahr | Support extern Jahr | Lizenzgebühr Jahr | Miete | Wartung | Schulung | Sonstiges |
| Primus Handwerk | | X | | | | | 1800.- € | 1792.- € | | 5000.- € (Downtime) | | |

Tabelle B.17: Workflows - Ausblick

B.4 Subversion Aktivitätsvergleich



(a) Compiere



(b) Adempiere

Abbildung B.2: Subversion Aktivitätsvergleich **Quelle: Sourceforge**

B.5 Modellierungen / Placeholder GmbH

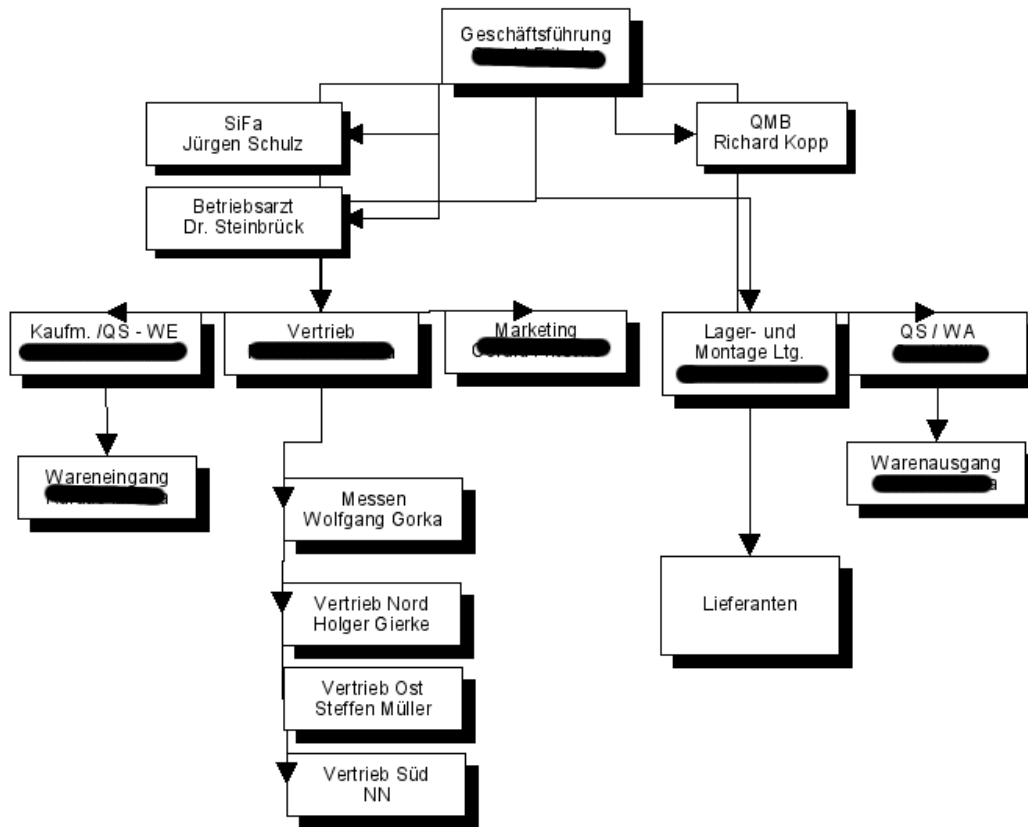


Abbildung B.3: Organigramm der Placeholder GmbH (vgl. Mueller und Kopp, 2007)

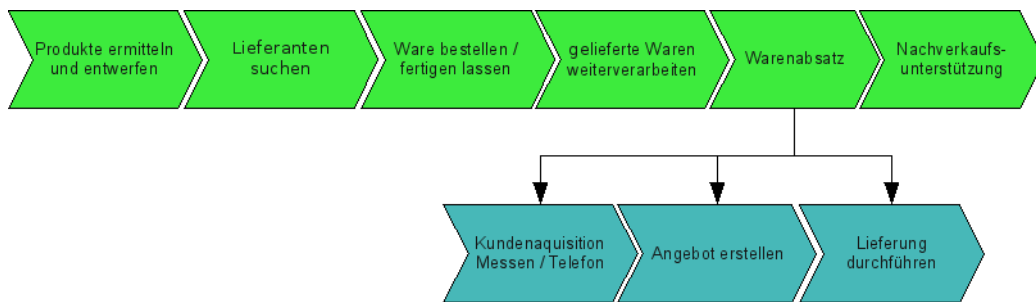


Abbildung B.4: Wertschöpfungskette der Placeholder GmbH

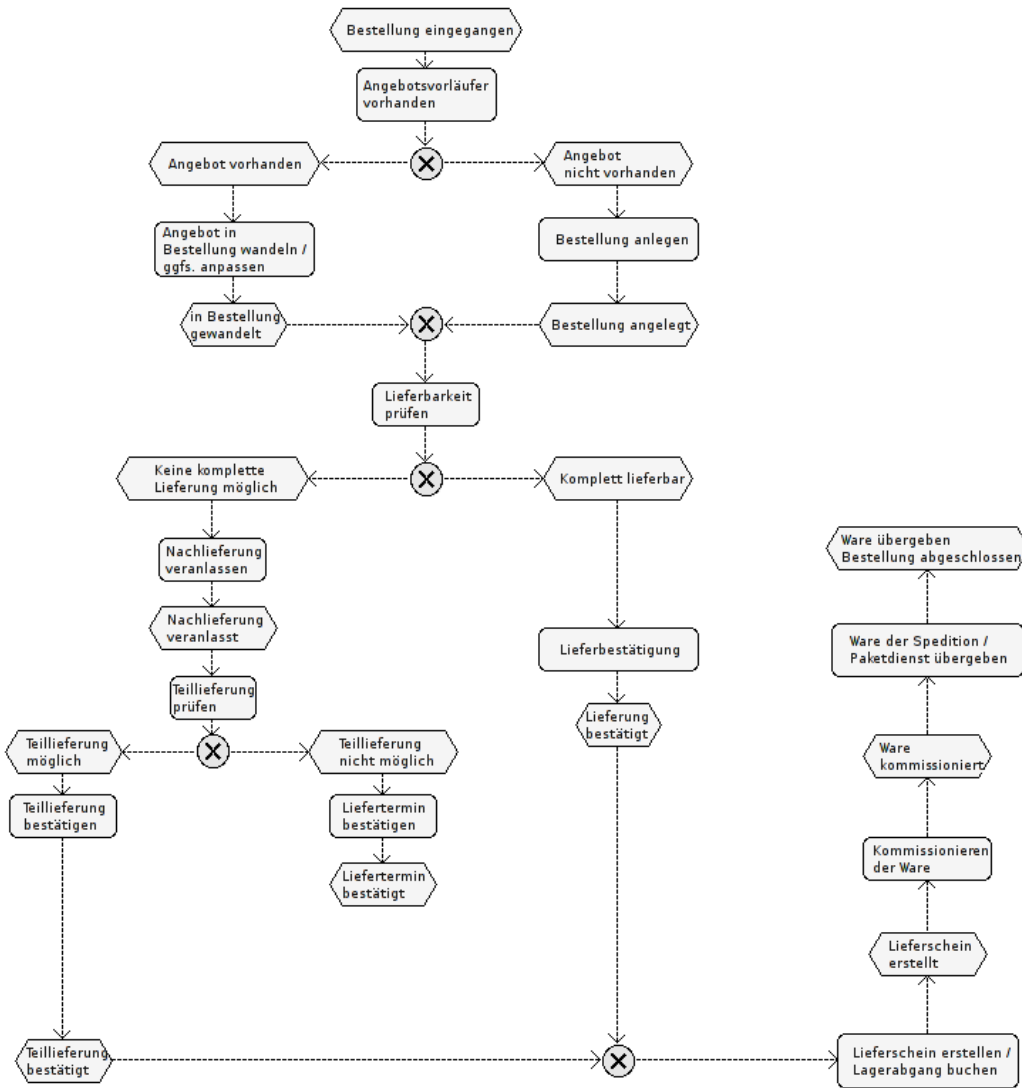


Abbildung B.5: Prozessmodell der Kundenbestellung bei der Placeholder GmbH

Kapitel C

Umfrage

C.1 Einleitung Umfrage

Im Rahmen dieser Arbeit wurde eine Umfrage mit dem Thema „Fragen zum Einsatz von Open Source Software in Unternehmen“ durchgeführt. Die Befragung wurde als Interview mit Fragebogen und parallel als Onlineumfrage durchgeführt. Praktische Erfahrung und Literaturquellen (vgl. z. B. Wichmann, 2004, S. 33) haben gezeigt, dass OSS häufig ohne das Wissen von Entscheidern und/oder notwendige analytische Verfahren zum Einsatz kommt. Dennoch ist ein anhaltend hohes Interesse an OSS erkennbar. Ein 2005 in den USA durchgeführte Studie zeigte, dass 87% aller dort befragten mittleren und großen Unternehmen OSS einsetzte (vgl. Walli et al., 2005, S. 4.).

C.2 Ziel der Umfrage

Die von IDC, Berlecon Research und Forrester Research in Europa und Optaros in den USA durchgeführten Studien konnten zeigen, dass OSS bei Infrastrukturdiensten weit verbreitet ist und dabei immer öfter auch im geschäftskritischen Bereich eingesetzt wird (vgl. Lykkegaard, 2005; Wichmann, 2002; Méndez et al., 2005; Walli et al., 2005). Hierbei stellt sich zwangsläufig die Frage, welche Maßstäbe und Verfahren im Rahmen der

unternehmerischen Evaluierung von OSS herangezogen werden und welche Gründe Unternehmen dazu bewegten, überhaupt über den Einsatz von OSS nachzudenken.

Die Umfrage erfüllt zwei Ziele. Zum einen soll sie die in den Studien ermittelten hohen Verbreitungswerte stützen, zum anderen folgende Eingangshypothesen prüfen:

1. Ersparnisse bei Lizenzkosten sind einer der wichtigsten Faktoren für den Einsatz von OSS.
2. Bei der Einsatzentscheidung zu OSS sind eher strategische Faktoren als analytische Verfahren ausschlaggebend.
3. In den meisten Fällen handelt es sich beim Einsatz von OSS um eine Migrationsentscheidung.

C.3 Vorgehen bei der Umfrage

Die Unternehmen wurden zufällig ausgewählt und telefonisch, persönlich oder mit Hilfe eines Internetfragebogens befragt. Der Fragebogen (vgl. Kramer, 2007) enthält vier Fragengruppen. Zunächst wurden Fragen zum Unternehmen gestellt, in dem der Befragte tätig ist. In Abhängigkeit von der Antwort folgten Fragen zum Einsatz von OSS im Unternehmen oder den Plänen des Unternehmens hinsichtlich des Einsatz von OSS. Die Umfrage ist angesichts der befragten 46 Studienteilnehmer nicht repräsentativ.

Von den Teilnehmern der Entscheidung wurde lediglich gefordert, dass sie an Softwareeinsatzentscheidungen beteiligt sind. An Unternehmensgröße, Mitarbeiterzahl oder Branchenzugehörigkeit wurden keine Voraussetzungen geknüpft.

C.4 Ergebnisse der Umfrage

C.4.1 Allgemeine Ergebnisse

Von insgesamt 46 befragten Experten entscheiden sechs alleine, 24 im Team und 16 als beratende Experten. Die Branchen, in denen die Befragten tätig sind, setzen sich wie folgt zusammen:

| Zusammensetzung nach Branche | | |
|---------------------------------------|---------|------------|
| Branche | absolut | prozentual |
| Automobil | 3 | 6.52% |
| Finanz / Versicherung | 3 | 6.52% |
| Handel | 1 | 2.17% |
| Konsumgüter | 1 | 2.17% |
| IT | 10 | 25.64% |
| Öffentlicher Dienst / Behörde | 8 | 20.48 % |
| Unternehmensberatung / Dienstleistung | 3 | 7.68% |
| Getränke | 1 | 2.17% |
| Sonstige | 16 | 34,72% |

Tabelle C.1: Umfrage - Branchenzusammensetzung

Neununddreißig der Befragten gaben an, OSS bereits im Unternehmen einzusetzen. Sieben Experten gaben an, OSS nicht im Unternehmen einzusetzen.

C.4.2 Unternehmen die OSS einsetzen

Die folgenden Statistiken geben lediglich den Teil der Befragungen wieder, der eine positive Antwort auf die Frage nach dem derzeitigen Einsatz von OSS ergab.

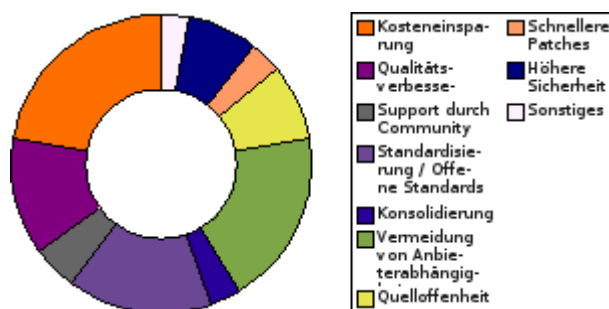


Abbildung C.1: Hauptgründe des Interesses an OSS

| Was sind die Hauptgründe ihres Interesses an Open Source ? | | |
|--|---------|------------|
| Grund | absolut | prozentual |
| Kosteneinsparung | 32 | 82.05% |
| Qualitätsverbesserung | 18 | 46.15 % |
| Support durch Community | 7 | 17.95% |
| Standardisierung / Offene Standards | 22 | 56.41% |
| Konsolidierung | 5 | 12.82% |
| Vermeidung von Anbieterabhängigkeiten | 27 | 69.23% |
| Quelloffenheit | 12 | 30.77% |
| Schnellere Patches | 5 | 12.82% |
| Höhere Sicherheit | 11 | 28.21% |
| Sonstiges | 4 | 10.26% |

Tabelle C.2: Umfrage - Hauptgründe des Interesses an OSS

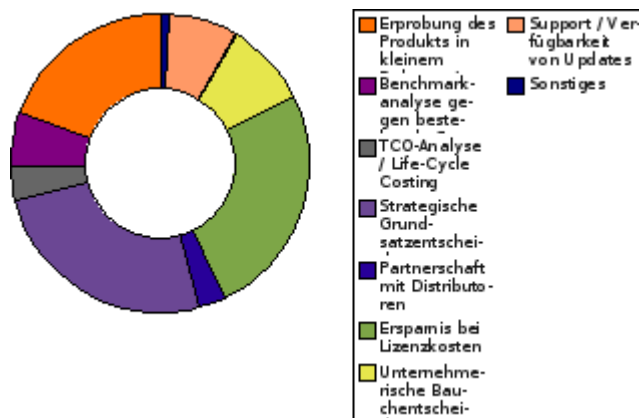


Abbildung C.2: Ausschlaggebende Entscheidungsfragen beim Einsatz von OSS

| Ausschlaggebende Entscheidungsfragen (jeweils die drei häufigsten) | | |
|---|---------|------------|
| Grund | absolut | prozentual |
| Erprobung des Produkts in kleinem Rahmen / kleinen Projekten | 21 | 53.85% |
| Benchmarkanalyse gegen bestehende Systeme | 6 | 15.38% |
| TCO-Analyse / Life-Cycle Costing | 4 | 10.26% |
| Strategische Grundsatzentscheidung | 27 | 69.23% |
| Partnerschaft mit Distributoren | 3 | 7.69% |
| Ersparnis bei Lizenzkosten | 27 | 69.23% |
| Unternehmerische Bauchentscheidung | 10 | 25.64% |
| Support / Verfügbarkeit von Updates | 8 | 20.51% |
| Sonstiges | 1 | 2.56% |

Tabelle C.3: Umfrage - Entscheidungsfragen

| Art des Einsatzes | | |
|--------------------------|---------|------------|
| Art | absolut | prozentual |
| Ersteinsatz | 13 | 33.33% |
| Substitution | 5 | 12.82% |
| Migration | 21 | 53.85% |

Tabelle C.4: Umfrage - Einsatzart

| Eingesetzte Softwareklasse | | |
|-----------------------------------|---------|------------|
| Art | absolut | prozentual |
| Basissoftware | 38 | 97.44% |
| Anwendungssoftware | 31 | 79.49% |

Tabelle C.5: Umfrage - Softwareklasse

| Eingesetzte Software | | | |
|-----------------------------|-------------------|-------------|-----------------|
| Name des Produkts | ist nicht bekannt | ist bekannt | wird eingesetzt |
| Linux | (0) 0.0% | (2) 5.13% | (37) 94.87% |
| FreeBSD | (8) 20.51% | (28) 71.79% | (3) 7.69% |
| Open Office | (0) 0.0% | (15) 38.46% | (24) 61.54% |
| Mozilla Firefox | (0) 0.0% | (4) 10.26% | (35) 89.74% |
| Samba | (0) 0.0% | (7) 17.95% | (32) 82.05% |
| MySQL | (1) 2.56% | (9) 23.08% | (29) 74.36% |
| PostgreSQL | (3) 7.69% | (20) 51.28% | (16) 41.03% |
| Mambo / Joomla | (15) 38.46% | (16) 41.03% | (8) 20.51% |
| Compiere / Adempiere | (23) 58.97% | (14) 35.90% | (2) 5.13% |
| TinyERP | (32) 82.05% | (7) 17.95% | (0) 0.0 % |
| Eclipse | (5) 12.82% | (11) 28.21% | (23) 58.97% |
| phpMyAdmin | (7) 17.95% | (12) 30.77% | (20) 51.28% |
| KeePass | (27) 69.23% | (7) 17.95% | (5) 12.82% |
| ClamAV | (20) 51.28% | (12) 30.77% | (7) 17.95% |
| Spamassassin/Amavis | (21) 53.85% | (13) 33.33% | (5) 12.82% |
| Nagios | (19) 48.72% | (4) 10.26% | (16) 41.03% |
| Open Reports | (34) 87.18% | (4) 10.26% | (1) 2.56% |
| Jasper Reports | (31) 79.49% | (7) 17.95% | (1) 2.56% |
| Mantis | (32) 82.05% | (4) 10.26% | (3) 7.69% |

Tabelle C.6: Umfrage - Software

| Wie wurden notwendige Analysen durchgeführt | | |
|--|---------|------------|
| Art | absolut | prozentual |
| selbst | 26 | 66.67% |
| extern | 2 | 5.13% |
| beides | 11 | 28.21% |

Tabelle C.7: Umfrage - Analysen

C.4.3 Unternehmen die aktuell keine OSS einsetzen

Die folgenden Statistiken geben lediglich den Teil der Befragungen wieder, der eine negative Antwort auf die Frage nach dem derzeitigen Einsatz von OSS ergab. Von den sieben Befragten, die in diese Gruppe fallen, gab lediglich einer an, dass im Unternehmen über den Einsatz von OSS nachgedacht wird (Grund: vermutete Lizenzkostensparnis). Die überwiegende Mehrheit von sechs Befragten gab an, in absehbarer Zeit keine OSS einsetzen zu wollen.

C.5 Zusammenfassung der Ergebnisse

Bei der durchgeführten Studie handelt es sich um eine Stichprobe. Die Stichprobe ist jedoch wegen ihrer vergleichsweise geringen Teilnehmerzahl nicht repräsentativ. Der Anteil der Unternehmen, die OSS einsetzen liegt laut Studie bei ca. 85%. In der Befragung sind, allerdings unbeabsichtigt, der öffentliche Sektor (20.48%) und die IT-Branche (25.64%) überrepräsentiert. Bezüglich der Eingangshypothesen zeigen die Ergebnisse, dass wie angenommen mit knapp 54% die Migration der häufigste Fall des Einsatzes eines neuen Anwendungssystems ist. Darüber hinaus ist laut dieser Stichprobe die vermutete Kostenersparnis für mehr als 82% der Befragten der gewichtigste Grund für ihr Interesse an OSS. Die Aussage der zweiten Hypothese, bezüglich strategischer Gründe versus analytischer Verfahren zur Rechtfertigung für OSS Projekte, lässt sich indes durch diese Stichprobe weder stützen noch widerlegen.

C.6 Kritische Würdigung der Ergebnisse

Die Umfrage ist nicht repräsentativ und kann daher weder Aussagen statistisch korrekt widerlegen noch bekräftigen. Dennoch stimmt das Ergebnis dieser Studie hinsichtlich des Einsatzes von OSS in Unternehmen (ca. 85% der befragten Unternehmen setzen OSS ein) mit dem von Walli et al. (vgl. Walli et al., 2005, S. 4.) nahezu überein. Die Umfrage wurde mit Beginn der Arbeit mit dem Ziel der zeitnahen Erhebung der IST-Situation durchgeführt.

Literaturverzeichnis

- Andreessen, M. (1998): *Netscape announces plans to make Next-Generation Communicator Source Code available free on the net.* Website. <http://wp.netscape.com/newsref/pr/newsrelease558.html> (besucht am 22.04.2007).
- Balzert, H. (1997): Lehrbuch der Software-Technik, Bd.2, Software-Management, Software-Qualitätssicherung und Unternehmensmodellierung, m. CD-ROM. Spektrum Akademischer Verlag.
- Bamberg, G. und Coenenberg, A. G. (2006): Betriebswirtschaftliche Entscheidungslehre (WiSo-Kurzlehrbücher). Vahlen.
- Bensberg, F., Grob, H. L. und vom Brocke, J. (2007): *Kostenorientiertes Controlling von E-Learning-Plattformen mit dem TCO-Konzept - Methodische Grundlagen, Softwareunterstützung und Entwicklungsperspektiven.* Website. <http://elead.campussource.de/archive/3/1080/> (besucht am 5.8.2007).
- Biethahn, J., Mucksch, H. und Ruf, W. (1994): Ganzheitliches Informationsmanagement 1. Grundlagen. Oldenbourg R. Verlag GmbH.
- Black, F. und Scholes, M. (1973): *The Pricing of Options and Corporate Liabilities.* Journal of Political Economy, Band 81, S. 637–654.
- Böcker, L. (2007): Open Source Jahrbuch 2007, Kapitel Die GPL v3 ein Schutzschild gegen das Damoklesschwert der Softwarepatente ?, S. 511–522. Lehmanns Media.
- Brügge, B., Harhoff, D. und Picot, A. (2004): Open-Source-Software. Eine ökonomische und technische Analyse. Springer, Berlin.

- Brynjolfsson, E. (1993): *The Productivity Paradox of Information Technology: Review and Assessment Communications of the ACM, December, 1993*. Technischer Bericht, MIT Sloan School of Management. <http://ccs.mit.edu/papers/CCSWP130/ccswp130.html> (besucht am 8.5.2007).
- Buschermöhle, R., Eekhoff, H. und Josko, B. (2006): *Erfolgs- und Misserfolgskriterien bei der Durchführung von Hard- und Softwareentwicklungsprojekten in Deutschland*. Technischer Bericht, VSEK / OFFIS e.v.
- Colony, G. F. (2007): *TEI For IT Organizations*. Technischer Bericht, Forrester Research. http://www.forrester.com/role_based/pdfs/teiforit_feb07.pdf (besucht am 29.06.2007).
- Comino, S. und Manenti, F. M. (2004): *Open Source vs Closed Source Software: Public Policies in the Software Market*. In 2nd IIO Conference (Chicago).
- Day, B. und Erickson, J. (2001): *Total Economic Impact™ Analysis*. Technischer Bericht, Giga Information Group. <http://h30097.www3.hp.com/pdf/compaq-tei-final-020125.pdf> (besucht am 29.06.2007).
- Eisendle, A. (2001): Lexikon der Wirtschaftsinformatik, S. 306. Springer Verlag.
- Ferstl, O. K. (2001): Lexikon der Wirtschaftsinformatik, S. 337–338. Springer Verlag.
- Fitzgerald, B. (2005): *Perspectives on Free and Open Source Software, Kapitel Has Open Source Software a Future*, S. 93–106. The MIT Press.
- Franck, E. und Jungwirth, C. (2001): *Open versus Closed Source*. Technischer Bericht, Institute for Strategy and Business Economics University of Zurich. ISSN 1660-1157.
- Freund, J. (2007): *Geschäftsprozesse vs. Workflows*. Website. <http://www.bpm-guide.de/articles/3> (besucht am 16.6.2007).
- Garvin, D. A. (1984): *What Does "Product Quality" Really Mean?*

- Gaulke, M. (2004): Risikomanagement in IT-Projekten. Oldenbourg.
- Ghosh, R. A. (2005): Perspectives on Free and Open Source Software, Kapitel Understanding Free Software Developers: Findings from the FLOSS Study, S. 23–46. The MIT Press.
- Ghosh, R. A. (2006): *Study on the: Economic impact of open source software on innovation and the competitiveness of the Information and Communication Technologies (ICT) sector in the EU*. Technischer Bericht, UNU-MERIT.
- Gonzales, A. R. (2001): *Department of Justice and Microsoft Corporation reach effective Settlement on Antitrust Lawsuit*. Technischer Bericht, Department of Justice. http://www.usdoj.gov/opa/pr/2001/November/01_at_569.htm (besucht am 21.04.2007).
- Görk, M. (2001): Lexikon der Wirtschaftsinformatik, S. 126–128. Springer Verlag.
- Haar, T. (2006): *Basel II wird Gesetz - IT-Sicherheit spielt größere Rolle bei der Kreditvergabe*. Website. <http://www.heise.de/newsticker/meldung/69680> (besucht am 13.07.2007).
- Hall, E. A. (2003): *Rapid Assessment for Distributed Computing*. Technischer Bericht, Gartner Group Inc. http://www.gartner.com/4_decision_tools/modeling_tools/costcat.pdf (besucht am 24.06.2007).
- Harker, P. T. und Vargas, L.-G. (1990): *Reply to "Remarks on the Analytic Hierarchy Process"*. Management Sciences, Band 36, Nr. 3, S. 270–273.
- Heise, A. (2003): *SCO verklagt IBM wegen Linux*. website. <http://www.heise.de/newsticker/meldung/35088> (besucht am 21.06.2007).
- Heise, A. (2004): *Berufungsgericht bestätigt Auflagen für Microsoft*. Website. <http://www.heise.de/newsticker/meldung/48760> (besucht am 07.08.2007).

- Heise, A. (2005): *Intel und Qinetiq wollen mit III-V-Halbleitern das Mooresche Gesetz fortführen*. website. <http://www.heise.de/newsticker/meldung/67115> (besucht 23.06.2007).
- Heise, A. (2007): *Wo das Geld herkommt*. Website. <http://www.heise.de/open/artikel/76859/2> (besucht am 21.04.2007).
- Hertzum, M. (1999): *Six Roles of Documents in Professionals Work*. Technischer Bericht, Centre for Human-Machine Interaction, Rise National Laboratory, Denmark. <http://www.ecscw.org/1999/03.pdf> (besucht am 02.08.2007).
- Herzog, T. (2006): *A Comparison of Open Source ERP Systems*. Technischer Bericht, Institute of Software Technology and Interactive Systems Business Informatics Group Vienna University of Technology. <http://mesh.dl.sourceforge.net/sourceforge/adempiere/ERPStudy.pdf> (besucht am 11.08.2007).
- Hnizdur, S. (2003): *The IDA Open Source Migration Guidelines*. Technischer Bericht, netproject Ltd. <http://ec.europa.eu/idabc/servlets/Doc?id=1983> (besucht am 04.05.2007).
- Hoegner, W., Hofmann, P. und Schießl, F. (2003): *Kurzfassung des Abschlussberichts inklusive Nachtrag. Client Studie der Landeshauptstadt München*. Technischer Bericht, Unilog Integrata GmbH. http://www.muenchen.de/cms/prod1/mde/_de/rubriken/Rathaus/40_dir/limux/publ/clientstudie_kurz.pdf (besucht am 5.6.2007).
- Jakob, S. (2006): *SOA löst das EAI-Versprechen ein*. IT-Director, Band 11. <http://www.it-director.de/startseite/itd-news/artikel/1186091961//353/soa-loest-das-eai-versprechen-ein.html> (besucht am 23.8.2007).
- Johnson, J. H. (1999): *Turning Chaos into Success*.
- Johnson, J. H. (2002): *It's a New World! And it's Called ROI!* Keynote der Chaos University 2002.

- Johnson, J. H. (2007): *2007 First Quarter Research Report*. Technischer Bericht, Standish Group Inc. http://www.standishgroup.com/quarterly_reports/pdf_copy/q1_2007_sample.pdf (besucht am 23.08.2007).
- Kellermann, J. (2005): *Peru: Freie Software per Gesetz*. Technischer Bericht, Bundestux.de. <http://www.bundestux.de/news/88304.html> (besucht am 24.04.2007).
- Klippstätter, K. (2004): *Blade-Rechner vor dem Durchbruch*. Website. http://www.computerwoche.de/knowledge_center/hardware/547189/ (besucht am 23.06.2007).
- Kofler, F. (2004): *Modelle zur Kostenbewertung von Open Source Projekten unter Berücksichtigung unterschiedlicher Umsetzungsgrade*. http://www.staedtebund.at/de/oss/referate/kofler_text.pdf (besucht am 24.06.2007).
- Koglin, O. und Metzger, A. (2004): Open Source Jahrbuch 2004, Kapitel Urheber- und Lizenzrecht im Bereich von Open-Source-Software, S. 293–304. Lehmanns Media.
- Kooths, S., Langenfurth, M. und Kalwey, N. (2003): *Open Source-Software Eine volkswirtschaftliche Bewertung*. Technischer Bericht, MICE Muenster Institute for Computational Economics. http://mice.uni-muenster.de/mers/mers4-OpenSource_de.pdf (besucht am 2.7.2007).
- Kramer, F. (2007): *Fragebogen zur Diplomarbeit*. Technischer Bericht, Otto-von-Guericke Universität.
- Krems, B. (2004): *Nutzwertanalyse*. website. <http://www.olev.de/n/nwa-kurz.htm> (besucht am 27.06.2007).
- Krishnamurthy, S. (2005): Perspectives on Free and Open Source Software, Kapitel An Analysis of Open Source Business Models, S. 279–296. The MIT Press.
- Kunath, C. (2005): Open Source Jahrbuch 2005, Kapitel Migration bei der BStU auf Linux-Netware/Windows xp, S. 25–36. Lehmanns Media.

- Langmack, J. (2006): *Migrationsleitfaden Version 2.1*. Technischer Bericht, Bundesministerium des Innern.
- Laube, H. (2007): *Novell gewinnt im Streit um Unix*. Financial Times Deutschland, Band 155/33, S. 4.
- Lee, J.-A. (2006): *New Perspectives on Public Goods Production: Policy Implications of Open Source Software*. <http://opensource.mit.edu/papers/LeeOSS.pdf> (besucht am 23.08.2007).
- Leipelt, D. (2000): *Grundgedanken zu Wirtschaftlichkeitsbetrachtungen für den Einsatz von Open Source Software*. Technischer Bericht, KBSt.
- Leiteritz, R. (2004): Open Source Jahrbuch 2004, Kapitel Open Source-Geschäftsmodelle, S. 139–170. Lehmanns Media.
- Leonard, A. (2000): *Do-it-yourself giant brains*. Website. http://archive.salon.com/tech/fsp/2000/06/22/chapter_2_part_two/print.html (besucht am 17.08.2007).
- Lerner, J. und Tirole, J. (2005): Perspectives on Free and Open Source Software, Kapitel Economic Perspectives on Open Source, S. 47–78. The MIT Press.
- Lutz, B., Potakowskyj, J., Richter, K., Starnberger, K. und Weissenberger, R. (2004): *Studie OSS. Open Source Software am Arbeitsplatz im Magistrat Wien*. Technischer Bericht, Stadt Wien, MA 14-ADV, Rathausstraße 1, 1082 Wien. <http://www.wien.gv.at/ma14/pdf/oss-studie-deutsch-langfassung.pdf> (besucht am 24.04.2007).
- Lykkegaard, B. (2005): *Western European End-User Survey*. Technischer Bericht, IDC.
- Mellis, W. (2001): Lexikon der Wirtschaftsinformatik, S. 422–423. Springer Verlag.
- Méndez, M. A., Reitsma, R. und Giordanelli, A. (2005): *Is Open Source Gaining Adoption In Europe?* Website. <http://www.forrester>.

- [com/Research/Document/Excerpt/0,7211,38061,00.html](http://www.com/Research/Document/Excerpt/0,7211,38061,00.html) (besucht 07.08.2007).
- Mertens, P., Bodendorf, F. und König, W. (2004): Grundzüge der Wirtschaftsinformatik (springer lehrbuch). Springer, Berlin.
- Meuer, H. (2006): *TOP500 Report for November 2006*. Technischer Bericht, Top500.org. http://www.top500.org/static/lists/2006/11/top500_statistics.pdf (besucht am 07.08.2007).
- Meyer, J. und Brunke, C. (2005): Open Source Jahrbuch 2005, Kapitel Migration auf Samba/Openldap bei der Norddeutschen Affinerie AG, S. 51–60. Lehmanns Media.
- Michlmayr, M. (2005): *Software Process Maturity and the Success of Free Software Projects*. In Software Engineering: Evolution and Emerging Technologies, S. 3–14. IOS Press, Krakow, Poland.
- Miller, G. A. (1956): *The magical number seven, plus or minus two: Some limits on our capacity for processing information*. Psychological Review, Band 63, S. 81–97. <http://www.musanim.com/miller1956/> (besucht am 5.8.2007).
- Moore, G. E. (1965): *Cramming more components onto integrated circuits*. <http://download.intel.com/research/silicon/moorespaper.pdf> (besucht 23.06.2007).
- Mueller, G. und Kopp, R. (2007): *Integriertes Management System Handbuch der Placeholder GmbH*. Technischer Bericht, Placeholder GmbH. IMSH-2006 (REV1).pdf.
- Mühlbauer, P. (2000): *Die Resozialisierung des Giganten*. Website. <http://www.heise.de/tp/r4/artikel/8/8778/1.html> (besucht am 07.08.2007).
- Nagel, K. (1990): *Nutzen der Informationsverarbeitung*.

- Netcraft (2007): *April 2007 Web Server Survey*. Technischer Bericht, Netcraft. http://news.netcraft.com/archives/web_server_survey.html (besucht am 07.08.2007).
- Neumann, P. G. (2005): Perspectives on Free and Open Source Software, Kapitel Attaining Robust Open Source Software, S. 123–126. The MIT Press.
- Norris, V. A. und Young, L. R. (2005): *Risk Assessment in Sarbanes Oxley*. Technischer Bericht, Advanced Technology Institute, Charleston SC; Bright Mind Solutions, Tampa FL.
- Otis, B., Flieger, K. und von Rotz, B. (2007): *Open Source Catalogue 2007*. Technischer Bericht, Optaros.
- Pape, U. und Merk, A. (2003): *Zur Angemessenheit von Optionspreisen*. Technischer Bericht, Europäische Wirtschaftshochschule Berlin. <http://www.escp-eap.de/upldata/Optionspreise2.pdf> (besucht am 29.06.2007).
- Perens, B. (2005): *The Emerging Economic Paradigm of Open Source*. Website. <http://perens.com/Articles/Economic.html> (besucht am 22.04.2007).
- Porter, M. E. (1998): On Competition (Harvard Business Review). McGraw-Hill Professional.
- Porter, M. E. (1999): *Competitive Strategy: Techniques for Analyzing Industries and Competitors*. Free Press.
- Rath, M. (2007): *Law and Order: Was ist IT-Compliance?* website. http://www.computerwoche.de/it_strategien/it_management/590497/index.html (besucht am 13.07.2007).
- Raymond, E. S. (2001): *The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary*. O'Reilly Media.

- Renner, T., Vetter, M., Rex, S. und Kett, H. (2005): *Open Source Software: Einsatzpotenziale und Wirtschaftlichkeit*. Technischer Bericht, Fraunhofer-Institut für Arbeitswirtschaft und Organisation IAO.
- Riklin, K. (2006): *FOSS-Studie Schweiz 2006*. Technischer Bericht, Schweizerischer Verband der Informations- und Kommunikationstechnologie SwisSICT.
- Rohr, T. (2004): Einsatz eines mehrkriteriellen Entscheidungsverfahrens im Naturschutzmanagement. Dissertation, Christian-Albrechts-Universität zu Kiel. http://e-diss.uni-kiel.de/diss_1108/d1108.pdf (besucht am 28.06.2007).
- Rosemann, M. (1996): *Komplexitätsmanagement in Prozeßmodellen*. Gabler Verlag.
- Ross, J. W. und Weill, P. (2006): *IT-Strategie: Die sechs wichtigsten IT-Entscheidungen*. <http://www.manager-magazin.de/it/artikel/0,2828,419482,00.html> (besucht am 09.06.2007).
- Röthig, D. P., Bergmann, K. und Müller, C. (2004): *WiBe 4.0 Empfehlung zur Durchführung von Wirtschaftlichkeitsbetrachtungen in der Bundesverwaltung, insbesondere beim Einsatz der IT*.
- Rüttgers, J. (2007): *FAQ: Was ist ein Fachverfahren ?* Website. http://www.service.nrw.de/faq/allgemeineFragen/faq_02/index.php (besucht am 15.6.2007).
- Rymer, J. R. und Cormier, B. (2003): *The Total Economic Impact™ of Developing and Deploying Applications on Microsoft and J2EE/Linux Platforms*. Technischer Bericht, Microsoft Corp. <http://download.microsoft.com/download/5/7/9/57996ae9-5024-464b-82e1-2154f1292d27/GIGA%20TEICaseStudy.pdf> (besucht 29.06.2007).
- Saaty, T. L. (1980): *The Analytic Hierarchy Process: Planning Setting Priorities, Resource Allocation*. McGraw Hill Higher Education.

- Saleck, T. und Safari-Fard, M. (2005): Chefsache Open Source. Vieweg.
- Sarbanes, P. S. und Oxley, M. (2002): *Sarbanes-Oxley Act of 2002*. Technischer Bericht, Congress of the United States of America. <http://www.kpmg.de/library/pdf/amtlicheFassung.pdf> (besucht am 15.05.2007).
- Schneider, A. (2007): *Die neue GPL: Änderungen und Entstehung im Überblick*. Website. <http://www.telemedicus.info/article/299-Die-neue-GPL-AEnderungen-und-Entstehung-im-UEberblick.html#extended> (besucht am 16.08.2007).
- Schröder, A. (2004): Open Source Jahrbuch 2004, S. 17–28. Lehmanns Media.
- Schumann, M. (2001a): Lexikon der Wirtschaftsinformatik, S. 332–333. Springer Verlag.
- Schumann, M. (2001b): Lexikon der Wirtschaftsinformatik, S. 251–252. Springer Verlag.
- Schumann, M. (2001c): Lexikon der Wirtschaftsinformatik, S. 504–505. Springer Verlag.
- Schwab, F. (2007): *IT-Strategie*. <http://www.4managers.de/themen/it-strategie/?type=123> (besucht am 09.06.2007).
- Schwarze, J. (2001): Grundlagen der Statistik, Bd.1, Beschreibende Verfahren (NWB-Studienbücher Wirtschaftswissenschaften). Verlag Neue Wirtschafts-Briefe.
- Seibt, D. (2001): Lexikon der Wirtschaftsinformatik, S. 46–47. Springer Verlag.
- Shankland, S. (1999): *Red Hat shares triple in IPO*. website. <http://news.com.com/2100-1001-229679.html> (besucht am 22.04.2007).
- Sherman, J. (2007): *Department of Justice, Antitrust Division Manual*. Website. <http://www.usdoj.gov/atr/foia/divisionmanual/two.htm#a1> (besucht am 12.04.2007).

- Stahlknecht, P. und Hasenkamp, U. (2005): Einführung in die Wirtschaftsinformatik. (springer lehrbuch). Springer, Berlin, 11. Auflage.
- Stahlknecht, P. und Nordhauss, R. (1981): Fallstudie Methodik der Hardware- und Software- Auswahl in kleinen und mittleren Unternehmen, dargestellt am Beispiel eines Fachverlages. Oldenbourg R. Verlag GmbH.
- Stallman, R. (2007): *The Free Software Definition*. Website. <http://www.fsf.org/licensing/essays/free-sw.html> (besucht am 22.06.2007).
- Stieler, W. (2002): *Unendliche Geschichte*. CT, Band 20, S. 3.
- Thilloy, C. (2006): *SOA in the Enterprise: A Survey of the Technical Landscape*. <http://www.soamag.com/I1/0906-1.pdf> (besucht am 30.6.2007).
- Vaughan-Nichols, S. J. (2004): *Get the FUD*. Website. <http://www.eweek.com/article2/0,1895,1426514,00.asp> (besucht am 07.08.2007).
- Vriendt, K. D. (2003): *Notes on the use of the Cost Comparison Spreadsheet*. Technischer Bericht, IDABC. <http://ec.europa.eu/idabc/servlets/Doc?id=1995> (besucht am 26.06.2007).
- Waldman, E. (2006): *InfiniBand Takes Market Share from Ethernet and Proprietary Interconnects*. Technischer Bericht, Mellanox Technologies. http://www.mellanox.com/pdf/presentations/Top500_Nov_06.pdf (besucht am 4.7.2007).
- Walli, S., Gynn, D. und von Rotz, B. (2005): *The Growth of Open Source Software in Organizations*. Technischer Bericht, Optaros.
- Werner, T. (2007): Open Source Jahrbuch 2007, Kapitel World domination - die Erfolgsgeschichte der Linux- und Open-Source Einführung im Auswärtigen Amt, S. 239–248. Matthias Bärwolff and Robert A. Gehring and Bernd Lutterbeck.
- Wheeler, D. A. (2007): *How to Evaluate Open Source Software / Free Software (OSS/FS) Programs*. Website. http://www.dwheeler.com/oss_fs_eval.html (besucht am 10.08.2007).

- Wichmann, T. (2002): *Use of Open Source Software in Firms and Public Institutions Evidence from Germany, Sweden and UK*. Technischer Bericht, Berlecon Research.
- Wichmann, T. (2004): *Linux- und Open-Source-Strategien*. (Xpert.Press). Springer, Berlin.
- Widmer, U. und Baehler, K. (2006): *Open Source Jahrbuch 2006, Kapitel Open-Source-Lizenzen Wesentliche Punkte für Nutzer, Entwickler und Vertreiber*, S. 163–179. Lehmanns Media.
- Wild, M. und Herges, S. (2000): *Total Cost of Ownership - Ein Überblick*. Technischer Bericht 1, Lehrstuhl für allg. BWL und Wirtschaftsinformatik, Johannes Gutenberg-Universität, Mainz.
- Wilde, K. (1992): *Controlling und EDV, Konzepte und Methoden für die Unternehmenspraxis*, S. 212–226. Huch, B; Behme, W; Schimmelpfeng, K.
- Windeck, C. (2007): *Schaulaufen. c't*, Band 6, S. 306.
- Wyssusek, B. (2001): *Lexikon der Wirtschaftsinformatik*, S. 210–211. Springer Verlag.

Abschließende Erklärung

Ich versichere hiermit, dass ich die vorliegende Diplomarbeit selbständig, ohne unzulässige Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Die aus fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht.

Magdeburg, den 01. September 2007