

# Open Source Software vs. Closed Source Software

Wie brauchbar sind Open Source Softwareprogramme  
für den Einsatz in Unternehmen?

MASTER THESIS

zur Erlangung des Grades „Master of Science“  
am Fachbereich Wirtschaftswissenschaften  
der Fachhochschule Liechtenstein, Vaduz  
(Bearbeitungszeitraum: 06.10.2003 – 17.09.2004)

eingereicht von:  
Herbert Janko, B.Sc.

Betreuer:  
Dr. Thomas Schake  
Dr. Friedrich Roithmayr

Siebenbrunnenfeldgasse 14/4  
A-1050 Wien

Wien, den 17. September 2004

## **Deutscher Abstrakt**

In den letzten Jahren rief Open Source Software (OSS) große Resonanz innerhalb und außerhalb der Computerindustrie hervor.

Es entstehen laufend neue Arbeiten (wissenschaftlich und nichtwissenschaftlich), wobei sich die meisten nur auf einzelne Gesichtspunkte beschränken. Meist ist dies die Beschreibung der Funktionalität der verschiedenen Softwarepakete oder Studien über Linux.

Diese Arbeit ist eine umfassendere Untersuchung von OSS im Unternehmensbereich, ohne auf Software Funktionalitäten einzugehen. Open Source Software hat eine lange Geschichte, jedoch erst in letzter Zeit reifte die Software, um für Unternehmen interessant zu werden. Dies zeigt sich vor allem in überdurchschnittlichen Zuwachsraten sowohl bei Unternehmen als auch bei Privatanwendern.

OSS hat es nicht einfach, gegen die vorhandenen proprietären Softwarelösungen im Unternehmensbereich zu konkurrieren. Open Source Software bietet einige Vorteile wie die kostenlose Verbreitung, was das Interesse vieler IT-Chefs geweckt hat.

Es kann nicht pauschal gesagt werden, dass OSS die gesamten IT-Kosten senken kann. Dies hängt von sehr vielen Faktoren ab, die in dieser Thesis näher erläutert werden.

Mit einer geeigneten Methodik wird untersucht, welche Einsparungspotentiale möglich sind und darauf aufbauend wird der am Besten geeignete Migrationsweg ermittelt.

## **English Abstract**

Open Source Software (OSS) had a huge response within and outside the computer industry in recent years.

Currently many articles (scientific and non-scientific) are written but most of them focus only on a few facts. They are written mostly about the functionality of the different software programs or about the Linux operating system.

This thesis is an all-embracing survey of OSS how it is used by companies but without response to functionality. Open Source Software has a long history however, it became interesting for companies to use this software in recent years. Hence, OSS has an above-average growth rate on the one hand in companies and on the other hand with private users.

It is not easy for OSS to compete against existing proprietary software solutions in departments. OSS has some advantages, e.g. free of charge, what makes it interesting for many IT-heads.

Globally could not be said that OSS reduces the costs of information technology. This depends on many facts, which are explained by this thesis.

All cost reduction possibilities are researched with the best method and furthermore the best possible migration is ascertained.

# Inhaltsverzeichnis

Abkürzungsverzeichnis .....	VII
Abbildungsverzeichnis .....	IX
Tabellenverzeichnis.....	X
1. Problemstellung und Zielsetzung.....	1
1.1 Motivation .....	1
1.2 Ziele .....	1
1.3 Problemabgrenzung .....	3
2. Open Source Software.....	4
2.1 Definition .....	4
2.2 Geschichte .....	6
2.3 Entwicklungsprozess bei OSS-Projekten .....	8
2.4 Open Source vs. proprietäre Software.....	10
2.4.1 Unterschiede in der Wertschöpfung .....	11
2.4.2 Unterschiede in der Entwicklung .....	13
2.4.3 Unterschiede in den Lizenzen .....	14
2.5 Open Source Software Projekte.....	15
2.5.1 Klassische Open Source Software .....	15
2.5.2 Kommerzielle Open Source Software.....	16
2.6 Rechtliche Aspekte .....	17
2.6.1 Allgemein .....	17
2.6.2 OSS Lizenzmodelle .....	19
2.6.3 Unterschiede zwischen den Lizenzmodellen.....	22
2.7 OSS Organisationen .....	22
2.7.1 GNU und die Free Software Foundation (FSF) .....	22
2.7.2 Open Source Initiative (OSI) .....	23
2.7.3 Apache Software Foundation (ASF) .....	23
2.7.4 Open Source Development Network (OSDN) .....	23
2.8 Distributionen .....	24
3. Ist-Situation.....	26
3.1 Verbreitung .....	26
3.2 Empfehlenswerte Einsatzgebiete .....	27

3.3 Marktsituation.....	28
3.3.1 Allgemeines .....	28
3.3.2 Produktmarkt .....	30
3.3.3 Dienstleistungsmarkt .....	37
3.3.4 Internetmarkt .....	39
3.4 SWOT Analyse .....	40
3.4.1 Grundlegendes .....	40
3.4.2 Stärken .....	41
3.4.3 Schwächen .....	43
3.4.4 Potenziale.....	45
3.4.5 Bedrohungen .....	47
4. Wirtschaftlichkeitsbetrachtung.....	50
4.1 Ausgangsszenario .....	50
4.2 Methodische Grundlagen .....	52
4.2.1 Allgemeines .....	52
4.2.2 Total Cost of Ownership.....	52
4.2.3 IT-Wirtschaftlichkeitsbetrachtung .....	54
4.2.4 Monetäre Analyse .....	54
4.2.5 Migration-Kosten-Matrix .....	55
4.2.6 Vergleichbarkeit .....	56
4.2.7 Neueinführung vs. Migration von Systemen .....	57
4.2.8 Vollkostenansatz .....	58
4.2.9 Nutzwertanalyse.....	59
4.3 Monetäre (operative) Dimension .....	60
4.3.1 Einsatzbereiche .....	60
4.3.2 Kostenkategorien.....	60
4.3.3 Eigenschaften der Unternehmensgrößen .....	61
4.4 Strategische Dimension .....	63
4.4.1 Makroökonomische Bedeutung .....	63
4.4.2 Mikroökonomische Bedeutung .....	64
4.5 Gesamtergebnis.....	64
5. Migration .....	67

5.1 Grundlegendes.....	67
5.2 Auswirkungen .....	68
5.3 Migrationswege .....	68
5.3.1 Schnelle Migration .....	68
5.3.2 Sanfte Migration .....	70
5.3.3 Kritische Erfolgsfaktoren.....	73
5.4 Migrationsarten.....	75
5.4.1 Grundlegendes .....	75
5.4.2 Vollständige Migration .....	77
5.4.3 Fortführende Migration .....	79
5.4.4 Teilweise Migration .....	79
5.5. Qualitativ-strategische Betrachtung .....	81
5.5.1 Grundsätzliche Betrachtung .....	81
5.5.2 Nutzwertanalyse.....	82
5.6. Migrationsempfehlungen .....	84
5.6.1 Weg der Entscheidungsfindung.....	84
5.6.2 Grundsatzempfehlungen.....	85
5.6.3 Vollständig „ablösende“ Migration .....	88
5.6.4 Vollständig „fortführende“ Migration.....	92
5.6.5 Punktuelle Teilmigration .....	95
5.6.6 Serverseitige Teilmigration .....	96
5.6.7 Gesamtergebnis .....	97
6. Fazit .....	98
Quellenverzeichnis.....	101
Eidesstattliche Erklärung .....	104
Anhang .....	105

## Abkürzungsverzeichnis

ASF	Apache Software Foundation
ASP	Application Service Provider
BIND	Berkeley Internet Name Domain
BMI	Bundesministerium der Innern (Deutschland)
BSD	Berkeley Software Distribution Unit
CASE	Computer Aided Software Engineering
CEO	Chief Information Officer
CGI	Common Gateway Interface
COLS	Commercial Linux Software
CSS	Closed Source Software
CUPS	Common Unix Printing System
DBMS	Database Management System
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name Service
DOS	Disk Operating System
FSD	Free Software Definition
FSF	Free Software Foundation
FTP	File Transfer Protocol
GIMP	GNU Image Manipulation Program
GNOME	GNU Network Object Model Environment
GNU	GNU is not Unix
GPL	GNU Public Licence
HTML	Hypertext Markup Language
HW	Hardware
i.d.R.	In der Regel
IP	Internet Protocol
IRC	Internet Relay Chat
ISP	Internet Service Provider
IT	Information Technology
KBSt	Koordinierungs- und Beratungsstelle der Bundesregie-

	<p>         rung für Informationstechnik          K Desktop Environment          Klein- und Mittelunternehmen          Lightweight Directory Access Protocol          Lesser GNU Public Licence          Mozilla Public Licence          Microsoft          Network File System          Open Source Definition          Open Source Development Network          Open Source Initiative          Open Source Software          Personal Computer          PHP: Hypertext Preprocessor          Reduced Instruction Set Computer          Storage Area Network          Source Code Management          Structured Query Language          Software          Strengths, Weakness, Opportunity, Threat          Total Cost of Ownership          User Help Desk          Visual Basic for Application          Virtual Private Network          Wirtschaftlichkeitsbetrachtung          World Wide Web          Intel-Plattform beginnend mit dem 8086 bis zum Pentium 4 und kompatible Prozessoren          Extended Markup Language       </p>
KDE	
KMU	
LDAP	
LGPL	
MPL	
MS	
NFS	
OSD	
OSDN	
OSI	
OSS	
PC	
PHP	
RISC	
SAN	
SCM	
SQL	
SW	
SWOT	
TCO	
UHD	
VBA	
VPN	
Wibe	
WWW	
x86	
XML	

# Abbildungsverzeichnis

Abbildung 1: Die klassische Wertschöpfung der Softwareindustrie .....	11
Abbildung 2: Veränderung der Wertschöpfung durch OSS.....	12
Abbildung 3: Struktur des IT-Marktes .....	28
Abbildung 4: Gesamter Open Source Unternehmensmarkt 2003.....	31
Abbildung 5: Installierte Server 2002 auf x86-Basis.....	32
Abbildung 6: Installierte Clients 2002 auf x86- Basis.....	33
Abbildung 7: Entwicklung des Weltmarkts für Embedded Linux, Software- Entwicklungstools und Dienste.....	34
Abbildung 8: Zusammenspiel von Marktplatz-Teilnehmern.....	40
Abbildung 9: Argumente für den Einsatz von OSS.....	43
Abbildung 10: Gegenargumente für den Einsatz von OSS .....	45
Abbildung 11: IT-WiBe-Methodik .....	54
Abbildung 12: Migrations-Kosten-Matrix mit Kostenkategorien und Einsatzfeldern.....	61
Abbildung 13: Sanfte Migration .....	71
Abbildung 14: Phasen der Umstellung bei einer sanften Migration.....	72
Abbildung 15: Modell: stufenförmiger Migrationsprozess .....	73
Abbildung 18: Entscheidungsprozess zur Einführung von OSS.....	85
Abbildung 19: Allgemeine Systemarchitektur mit einem linuxbasierten Fat Client .....	89

## Tabellenverzeichnis

Tabelle 1: Vergleich der Eigenschaften wichtiger OSS-Softwarelizenzen	.22
Tabelle 2: OSS-Produktmarktsegmente und deren Relevanz im kommerziellen Bereich.....	30
Tabelle 3: Der Markt für Linux-Dienstleistungen .....	38
Tabelle 4: Vergleich der benutzerbezogenen Migrationskosten für vollständige/fortführende Migration .....	76
Tabelle 5: Verteilung der Kosten bei "Vollständiger Migration" in Behörden .....	77
Tabelle 6: Gesamt-Migrationskosten je User bei vollständiger Migration	.78
Tabelle 7: Gesamt-Migrationskosten je User bei fortführender Migration.	79
Tabelle 8: Gesamt-Migrationskosten je Benutzer bei punktuellerer Migration .....	79
Tabelle 9: Migrationskostenverteilung .....	80
Tabelle 10: Gesamt-Migrationskosten je Benutzer bei serverseitiger Teilmigration .....	80
Tabelle 11: Migrationskostenvergleich vollständige und serverseitige Migration .....	81
Tabelle 12: Kriterienkatalog für die qualitativ-strategische Betrachtung..	83

# **1. Problemstellung und Zielsetzung**

## **1.1 Motivation**

In letzter Zeit wurde viel über den Einsatz von Open Source Software (kostenlose Software mit freiem Zugang zum Quellcode) in Behörden bzw. Unternehmen gesprochen – z. B.: in München, Brasilien, Asien,... - um nur einige wenige zu nennen. Vor allem dann, wenn veraltete Hard- und Software auf den neuesten Stand gebracht werden soll. Weiters erscheint fast monatlich eine neue Untersuchung von Marktforschungsunternehmen zu diesem Thema, jedoch immer mit dem Ergebnis, dass die auftraggebende Firma die besten Produkte entwickelt, d.h. wenn Microsoft eine Studie in Auftrag gibt, dann sind Microsoft Produkte die Besten und vice versa bei Open Source Unternehmen.

Ein Wechsel von kommerzieller Software (mit Lizenzgebühren und ohne frei zugänglichem Code) zu Open Source Software beeinflusst die zukünftige Unternehmensstrategie im IT-Bereich, weshalb eine genauere, wissenschaftliche Betrachtung immer wichtiger wird; genau da soll diese Master-Thesis ansetzen.

Es ist notwendig, die Chancen und Risiken, die Vor- und Nachteile beim Einsatz von Open Source Software in Unternehmen wissenschaftlich und objektiv zu untersuchen.

Eine weitere Motivation für diese Master-Thesis ist, dass der Autor dieser Master-Thesis durch seine Tätigkeit als Softwareentwickler auch Erfahrungen mit einigen kostenlosen Produkten machen konnte.

## **1.2 Ziele**

Vor dem Hintergrund der zunehmenden Diskussion über Kosteneinsparungen im IT-Bereich eines Unternehmens soll untersucht werden, ob sich durch die Auswahl von bzw. den Umstieg auf Open Source Software Kosteneinsparungspotenziale ergeben. Nebenbei soll auch noch untersucht

werden, ob nicht der Einsatz von Application Service Providern<sup>1</sup> (ASP) noch kostengünstiger käme.

Das Hauptaugenmerk dieser Master-Thesis soll auf die Möglichkeiten bzw. Einschränkungen, welche Open Source Software Produkte im Gegensatz zur kommerziellen Software heutzutage bieten, gelegt werden. Es soll den Entscheidungsträgern in den Unternehmen als Grundlage für ihre zukünftigen strategischen Entscheidungen im IT-Bereich dienen. Es wird untersucht, ob Unternehmen mit Open Source Software das Auslangen finden und in welchen Bereichen man Open Source Software nicht gebrauchen kann und Lizenz Software einsetzen muss.

Hierzu wird ein Kriterienkatalog erstellt, um die unterschiedlichen Software Typen vergleichbar zu machen. Besonders auf die Unangreifbarkeit des Endergebnisses wird geachtet.

Mögliche Kriterien sind:

- Total Cost of Ownership (TCO)
- Betriebskosten
- Wechselkosten
- Einhaltung von Standards
- Auswirkungen auf das Unternehmen (Soft- und Hard Facts)
- Stabilität (Release Häufigkeit)
- Sicherheit (auch vor Internet-Viren, -Würmern und trojanischen Pferden)
- Wartbarkeit
- Verfügbarkeit von deutschsprachigen Anwendungen
- usw.

Nicht zu vergessen ist eine Untersuchung, ob der derzeitige Boom bei Open Source Software mit dem Hype bei Internet-Firmen (New Economy) vor einigen Jahren vergleichbar ist und somit die gleichen Gefahren eines Zusammenbruchs (vgl. Dot-Com Krise) beinhaltet.

---

<sup>1</sup> Mehr Informationen über ASP bietet die Bachelorthesis von Janko (2002)

### **1.3 Problemabgrenzung**

Es ist unmöglich, alle Unternehmenstypen und deren Softwareeinsatz hinsichtlich Eignung für Open Source Produkte zu untersuchen, somit muss man Einschränkungen diesbezüglich treffen. Diese Master-Thesis soll sich vor allem auf Klein- und Mittelunternehmen (KMU's) beziehen, welche oftmals keine eigenen IT-Abteilungen haben und dadurch auf externe Berater angewiesen sind.

Die Eingrenzung auf einen Unternehmenstyp dient auch dazu, die Anzahl der Produkte zu beschränken, denn es gibt mehr als 100 000 Projekte, welche Open Source Software Produkte entwickeln. Sollte sich für eine Aufgabenstellung im Unternehmen kein brauchbares Open Source Produkt finden, so sollen auch verfügbare Freeware Produkte (kostenlos, jedoch ohne Zugang zum Code) untersucht werden.

Des Weiteren erfolgt keine detaillierte Beschreibung der einzelnen Produkte (d.h. es wird kein Benutzerhandbuch für das jeweilige Produkt erstellt) und auf die rechtliche Situation – Lizenzmodelle – wird, wenn überhaupt, nur am Rande eingegangen. Ebenso wird das ASP-Modell nicht detailliert beschrieben, da dies nicht Thema dieser Master-Thesis ist.

Weiters ergeben sich Einschränkungen hinsichtlich des Tauglichkeitstestes (Funktionalität) der Software, denn kaum ein Unternehmen wird freiwillig bereit sein, von heute auf morgen auf Open Source Produkte umzustellen und diese dann für diese Thesis im Unternehmenseinsatz zu testen. Somit muss man sich hierbei auf die Angaben der Programmierer auf den jeweiligen Homepages, auf eigene Erfahrungen und auf Laborversuche verlassen.

Da die Softwareprodukte laufend weiter entwickelt werden, ist es schwierig die Funktionalität zu vergleichen. Da die Master-Thesis in erster Linie ein allgemeingültiges Ergebnis liefern soll, wird auf die Beschreibung der Funktionalität verzichtet und es wird diesbezüglich auf die entsprechenden Fachzeitschriften oder Produkthomepages verwiesen.

## 2. Open Source Software

Am Beginn dieser Master-Thesis wird ein Überblick über Open Source Software (OSS) gegeben, welcher für das Verständnis der weiterführenden Analyse notwendig ist.

Folgende Fragen sollen geklärt werden:

- Was ist Open Source Software?
- Was sind die wesentlichen Unterschiede zwischen Open Source und proprietärer Software?
- Welche wichtigen Lizenzmodelle bestimmen den rechtlichen Aspekt von OSS?

### 2.1 Definition

Unter Open Source Software versteht man Software, deren Quellcode (= Source Code), also die geschriebenen Anweisungen des Programmierers, im Gegensatz zu herkömmlicher Software (= „proprietärer“ oder „Closed Source Software (CSS)“) frei zugänglich und veränderbar ist.

Die wesentlichen Eigenschaften von OSS sind:<sup>2</sup>

- Jeder hat das Recht, die Software nach eigenem Ermessen zu nutzen.
- Der Quelltext muss jedem Benutzer offen gelegt werden oder es muss auf eine frei zugängliche Stelle verwiesen werden, wo er erhältlich ist.
- Der Benutzer hat das Recht, die Software zu modifizieren und in modifizierter Form weiterzuverteilen.
- Die Lizenz darf niemanden im Verkauf oder der Weitergabe der Software in Form einer Softwarezusammenstellung einschränken.

---

<sup>2</sup> Vgl. Opensource (10.09.2004)

Im Gegensatz hierzu gilt für proprietäre Software:

- Vervielfältigung, Weiterverbreitung und Modifizierung sind untersagt.
- Das Nutzungsrecht wird in Form einer Lizenz erteilt.
- Eigentümer der Software ist nicht der Anwender, sondern weiterhin der Hersteller, denn er hat das Urheberrecht und die vollständige Kontrolle über das Produkt.

Mit der Aussage über die Quellcodeverfügbarkeit wird noch keine definitive Aussage darüber gemacht, ob eine Software kostenpflichtig ist. OSS kann sowohl kostenlos als auch entgeltpflichtig sein. Das gleiche gilt auch für proprietäre Software (z. B. Freeware ist proprietär und kostenlos)<sup>3</sup>.

Im Folgenden werden die beiden Begriffe „Open Source Software (OSS)“ und „proprietär“ als Beschreibung für Software mit offenem und verborgenem Quellcode benutzt. Wenn zusätzlich eine Aussage über den Kostenaspekt der Software getroffen werden soll, wird der Begriff „kommerziell“ verwendet.

Neben dem Begriff „Open Source Software“ wird auch der verwandte Begriff „Free Software“ (Freie Software) gebraucht, wobei frei im Sinne von Freiheit und nicht von kostenlos gemeint ist. Der Unterschied zwischen den Begriffen ist, dass OSS eher technisch orientiert ist, wohingegen der Begriff der „Free Software“ breiter angelegt ist. Er stellt die Freiheit des Nutzers im Umgang mit der Software stärker in den Vordergrund und ist auch philosophisch und gesellschaftspolitisch motiviert.

In dieser Arbeit wird durchgängig der Begriff der OSS als Überbegriff für alle Arten von Software mit offenem Quellcode verwendet, da er ideologiefreier und im kommerziellen Bereich gebräuchlicher ist.<sup>4</sup>

---

<sup>3</sup> Eine umfassendere Abgrenzung ist unter GNU Projekt (29.07.2001) zu finden

<sup>4</sup> Vgl. Leiteritz (2002), S. 4 f

## 2.2 Geschichte

Um das Verständnis für OSS und die Entwicklung dieser Software zu schaffen, ist es wichtig die Geschichte von OSS zu kennen. Hier wird nur ein kurzer Überblick vermittelt, da eine umfangreiche Darstellung der Geschichte nicht im Rahmen dieser Master-Thesis behandelt wird.<sup>5</sup>

Die Anfänge von Open Source Software gehen auf die Anfänge der Computerindustrie zurück.

In den Jahren 1960 bis 1970 bestand das Computergeschäft fast ausschließlich aus dem Verkauf und der Wartung von Hardware. Die Software wurde oft in Eigenregie entwickelt.

Mit der Entwicklung von Unix (1969) und der hohen Entwicklungstätigkeit an den Universitäten (insbesondere an der Universität von Berkeley, Kalifornien) begann der Siegeszug der Software.

Als dann 1982 die ersten kommerziellen Unix-Versionen auf den Markt kamen, wurde die liberale Lizenzpolitik und das Austauschen von Programmen Zug um Zug geändert.

Um dieser Entwicklung entgegen zu wirken, wurde von Richard Stallman das GNU-Projekt (GNU is not Unix) mit dem Ziel geschaffen, ein freies Unix zu entwickeln. Um alle Projekte zu koordinieren und die Einnahmen zu verwalten wurde 1984 die Free Software Foundation (FSF) gegründet. Stallman betrachtete es als natürliches Recht, seine Programme mit Kollegen zu teilen, vor allem da sich Software fast ohne Aufwand und mit marginalen Kosten vertreiben lässt. Durch die immer restriktiver werdenden Lizenzen der kommerziellen Softwareanbieter sah sich Stallman in diesem bis dahin üblichen Umgang mit Software gehindert.

Somit entwickelte Stallman mit der GNU General Public License (GPL) die erste Lizenz für OSS. Der darin auftauchende Begriff „Copyleft“ stellt sicher, dass Software die ihm unterliegt frei bleibt, d. h. jeder, der sie benutzt und weiterentwickelt, hat die gleichen Rechte wie der ursprüngliche Autor. Dieses Ziel stellt die GPL durch die Forderung sicher, dass jedes

Programm, das auch nur eine Zeile Code enthält, der der GPL unterliegt, wieder der GPL unterliegen muss.

Diese Eigenschaft macht die GPL nur schwer geeignet für die Entwicklung kommerzieller Software, da sie mit jeder anderen Lizenz, die mehr Einschränkungen enthält, inkompatibel ist.

Daher beschränkte sich OSS in den ersten Jahren ausschließlich auf den privaten und universitären Bereich.

Als 1991 das Linux-Projekt von Linus Torvalds gegründet wurde, gab es zum ersten Mal eine freie Alternative zu proprietären Betriebssystemen. Dadurch wurde die OSS-Entwicklung beschleunigt und immer mehr Entwickler sprangen auf diesen Zug auf.

Am Beginn war die OSS Entwicklung vor allem durch Lösungen für eigene Probleme geprägt. Erst 1998 erregte Netscape Aufsehen, durch die Freigabe des damals weit verbreiteten Browser „Netscape Navigator“ als OSS. Diesem Schritt folgten in den letzten Jahren weitere Unternehmen (z.B: Sun -> OpenOffice.org & StarOffice) was zum stürmischen Wachstum bei OSS beitrug und für viele Projekte den Bekanntheitsgrad erheblich erhöhte. Ab 1999 sind viele Projekte entstanden, OSS sowohl für Privatcomputer als auch für Rechenzentren in Unternehmen einsetzbar zu machen. Mittlerweile – 2004 - findet man für die meisten proprietären Softwareprodukte ein frei verfügbares Gegenstück, das eine ähnliche Funktionalität aufweist.<sup>6</sup>

*„Die weltweite Software-Industrie wird von amerikanischen Unternehmen dominiert. OSS ist dagegen ein primär europäisches Phänomen. Der Ursprung von Linux liegt mit Linus Torvalds in Finnland. Die Mehrzahl der an OSS mitarbeitenden Entwickler stammen aus Europa (53,7%), wobei Deutschland eine führende Position in Europa einnimmt (21,2% aller Entwickler weltweit kommen aus Deutschland) (Robles 2001). Auch der Verbreitungsgrad von Linux ist nirgendwo höher als in Europa.“<sup>7</sup>*

---

<sup>5</sup> Für tiefergehende Informationen werden einschlägige Bücher oder Internet Seiten empfohlen

<sup>6</sup> Vgl. Leiteritz (2002), S. 5 f, Widmer (2003), S. 5

<sup>7</sup> Leiteritz (2002), S. 6

## 2.3 Entwicklungsprozess bei OSS-Projekten

Nachdem nun einiges über die Geschichte von OSS geschrieben wurde, wird in diesem Abschnitt kurz berichtet, wie die Projekte zustande kommen und abgewickelt werden.

So wie die Entwicklung proprietärer Software ist auch die Entwicklung von Open Source Software ein komplizierter und schwer vorhersehbarer Prozess.

Umfangreiche Softwareprodukte sind komplex und daher fehlerbehaftet, teuer und sollten innerhalb eines definierten Zeitplanes fertig werden. Somit ist es erstaunlich, dass OSS-Projekte es schaffen, Millionen-Zeilen-Softwareprodukte zu erstellen, zu pflegen und zu erweitern. Dies meist ohne einer Firma im Hintergrund und alles in der Freizeit der jeweiligen Entwickler.

Bei OSS-Projekten gibt es eine Projektleitung, die nur wenige Personen umfasst. Diese managt sowohl die Verwaltung der Software als auch die Koordination zwischen den Entwicklern und trifft die Entscheidungen, welche Programmteile zu der bestehenden Software hinzugefügt werden. Sie vertreten das Projekt nach außen als Ansprechpartner, geben die Entwicklungsrichtung vor und sind Konfliktlöser bei Problemen innerhalb der Entwicklergemeinde. Jedoch sind die meisten OSS-Projekte nicht erfolgreich, so dass nur ein Bruchteil aller gestarteten Open Source Projekte es wirklich schafft, ein brauchbares Produkt zu entwickeln und hinreichend neue Entwickler und Anwender zu begeistern. Die sich durchsetzenden Projekte aber schaffen es unter anderem deshalb, weil sie von sehr guten Entwicklern getragen werden.

Im Gegensatz zu Unternehmen ist es für die Projektleitung einfach, motivierte Mitarbeiter zu finden. In OSS-Projekten entscheidet jeder Mitarbeiter selbst, woran er arbeiten will. Die Entwickler muss man nicht beaufsichtigen oder herumkommandieren, sondern sie entwickeln die Programmteile aus reiner Leidenschaft am Programmieren. Jeder beschäftigt sich mit dem Teilbereich, in dem er sich am Besten auskennt. Jedoch muss jeder Programmierer zuerst mit persönlicher Leistung die Projektlei-

tung überzeugen, um als Projektmitarbeiter arbeiten zu dürfen. Ein OSS-Projektteam besteht nicht nur aus Programmierern sondern auch aus Testern, Grafikdesignern, technischen Schreibern für die Dokumentation und Übersetzern, die die Software und die Dokumentation an verschiedene Sprachen anpassen.

Die Planung der notwendigen Releases erfolgt durch die Projektleitung in einem so genannten Reifezyklus. Die Software „reift“ von der ersten Veröffentlichung als Alpha-Version (neue Programmteile kommen hinzu) über diverse Beta-Versionen (bestehende Software wird fehlerbereinigt) bis hin zum fertigen Produkt (= Gamma-Version). Bei großen Projekten werden zusätzliche Meilensteine definiert, an denen die vorhandenen Programmteile zu einer Version zusammengefügt und veröffentlicht werden, um von den Anwendern getestet werden zu können.

In dieser Testphase wird versucht, eine möglichst breite Anwendergruppe zu finden, die durch Rückmeldungen helfen, die Fehler auszubessern.<sup>8</sup>

Manchmal sind sich die Entwickler nicht einig über die weitere Entwicklung des Projekts oder sie stimmen mit den Vorgaben der Projektleitung für die weitere Entwicklung nicht überein. Dann kann sich ein Teil der Projektgruppe als eigenes Projekt abspalten und konkurriert mit dem alten Projekt. Dieses Abspalten wird „Forking“ genannt und gilt als Schreckensszenario innerhalb der Entwicklergemeinde, weil zwei kleinere Teams als weniger produktiv angesehen werden, als ein größeres Team.

Dabei kann das Abspalten sowohl positive als auch negative Eigenschaften haben. Es können unter anderem angestaubte Programme einer Art „Kur“ unterzogen und somit grundlegend erneuert werden.

---

<sup>8</sup> Vgl. Ettrich (2004), S. 179 ff

Je größer das Entwicklerteam wird, umso wichtiger wird eine effektive und effiziente Kommunikation. Bei Open Source Projekten besteht die Kommunikation aus zwei Säulen: Dem Internet als schnelle und billige Möglichkeit, Daten auszutauschen und der englischen Sprache als international akzeptiertes menschliches Sprachprotokoll. Englisch hat sich als führende Sprache entwickelt, weil es international der kleinste gemeinsame Nenner ist und sich mit nationalen Sprachen kaum eine kritische Masse an Programmierern finden lässt.

Als technisches Hilfsmittel zur Zusammenarbeit wird das Internet eingesetzt. Die meist verwendeten Werkzeuge sind:

- E-Mail – persönliche Kommunikation und Mailinglisten
- Newsgroups – Diskussionsforen
- Internet Relay Chat (IRC) – direkte Diskussion der Mitarbeiter
- World Wide Web (WWW) – Information für das Projekt auf einer Webseite
- File Transfer Protocol (FTP) – Bereitstellung von neuen Versionen
- Source Code Management (SCM) – zentrales Depot für den gesamten Code eines Projekts
- Infrastrukturservices – Speicherplatz, Rechnerleistung und Netzwerk-Bandbreite<sup>9</sup>

## **2.4 Open Source vs. proprietäre Software**

Open Source Software unterscheidet sich in einigen Punkten wesentlich von proprietärer Software. Für das genauere Verständnis von OSS sollen in diesem Kapitel die Unterschiede zwischen den beiden Softwarekonzepten erläutert werden. Da die Funktionalität meistens sehr ähnlich ist und sich laufend (teilweise täglich) ändert, wird auf diesen Unterschied hier nicht eingegangen. Die wichtigsten Unterschiede zwischen den beiden Softwarekonzepten werden in den nächsten drei Abschnitten behandelt.

---

<sup>9</sup> Vgl. Ettrich (2004), S. 182 ff

## 2.4.1 Unterschiede in der Wertschöpfung

Wie im Kapitel über die Geschichte von OSS bereits erwähnt, war am Beginn der Computerindustrie die Software eher ein Nebenprodukt oder wurde selbst entwickelt.

Mit dem Aufschwung der Personal Computer (PC) verbreiteten sich einheitliche Betriebssysteme (z.B: DOS, Windows, MacOS), welche immer leistungsfähiger wurden. Durch diese quasi Standards wurde der Software-Markt immer wichtiger und Tausende von Applikationen entstanden mit der Zeit.

Es entwickelte sich eine völlig neue Industriebranche mit einer eigenen Wertschöpfungskette. Neben der Forschung, Entwicklung und Implementierung der Software waren noch weitere Aufgaben wie Beratung, Unterstützung und Anpassung auf Kundenwünsche wichtig.

Die folgende Abbildung zeigt die klassische Wertschöpfungskette der Softwareindustrie:

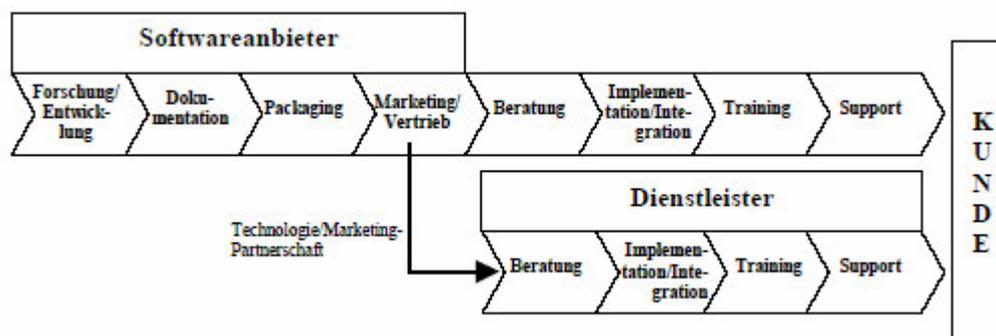


Abbildung 1: Die klassische Wertschöpfung der Softwareindustrie (aus: Leiteritz (2002), S. 8)

Bis zum Aufkommen von OSS wurde Software entweder von Unternehmen oder Universitäten entwickelt. Seit OSS eine breitere Masse an Entwicklern begeistert, veränderte sich auch das Innovationsoligopol. Da Open Source Software kostenlos erhältlich ist und auch der Source Code frei verfügbar ist, wurde der Prozess der Softwareentwicklung auf mehrere Akteure verteilt und somit offener, transparenter und vielfältiger.

Die folgende Abbildung zeigt den veränderten Wertschöpfungsprozess:

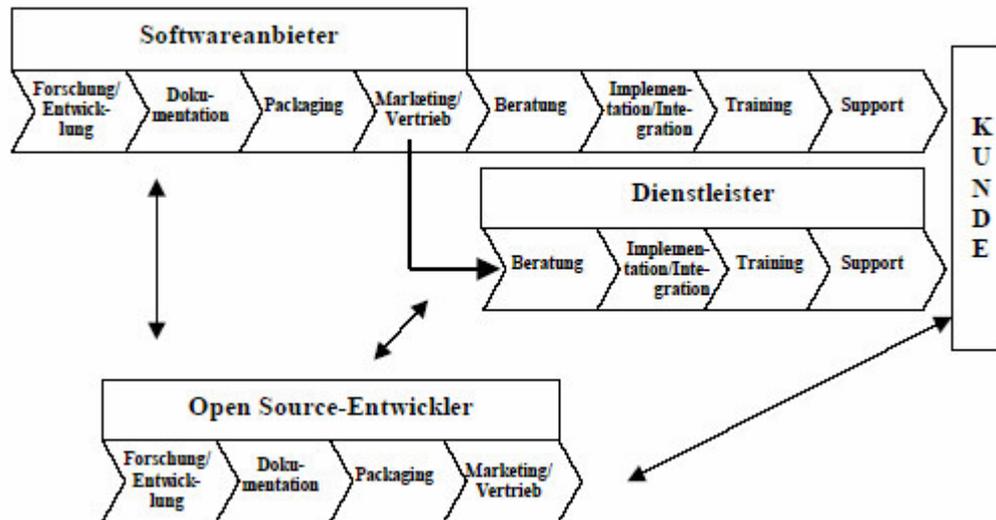


Abbildung 2: Veränderung der Wertschöpfung durch OSS

(aus: Leiteritz (2002), S. 8)

Nicht nur die Softwareentwicklung änderte sich, auch die Wertschöpfungskette wird durch den offenen Quellcode verändert:

- Dienstleister können Veränderungen am Quellcode vornehmen und auf individuelle Kundenbedürfnisse anpassen.
- Softwareanbieter müssen Software nicht von Grund auf selber entwickeln, sondern können auf eine OSS-Basis zurückgreifen.
- Nicht zuletzt kann auch der Kunde selbst das Endprodukt beeinflussen und erhält dadurch eine größere Unabhängigkeit.<sup>10</sup>

Howe prognostizierte die Veränderung der Softwarewertschöpfung durch die OSS-Entwicklung wie folgt:

*„Proprietary software vendors will suddenly see software development as an unfair fight: their mercenary band of captive developers against a battalion of Internet-armed revolutionaries“<sup>11</sup>*

<sup>10</sup> Leiteritz (2002), S. 7 ff

<sup>11</sup> Delio (2000)

## 2.4.2 Unterschiede in der Entwicklung

Die proprietären Softwareprodukte werden in IT-Unternehmen entwickelt. Dabei wird ein interner, zentralisierter und organisierter Entwicklungsprozess verfolgt und die entsprechende Infrastruktur genutzt.

Wie im Kapitel über die Entstehung von OSS-Projekten<sup>12</sup> bereits erwähnt, verfügt das OSS-Phänomen über andere Entwicklungsstrukturen. Das Überraschende ist, dass mittels OSS-Projekten eine vergleichbare oder manchmal sogar eine bessere Software entwickelt wird.

Ebenso unterschiedlich gestaltet sich die Testphase. Während es in proprietären Softwareunternehmen eine klare Trennung zwischen Entwicklungsphase und Testphase gibt, ist bei OSS jeder Entwickler sowohl Programmierer als auch Tester. Hinzu kommt, dass durch die Quelloffenheit die Möglichkeit besteht, Struktur- und Denkfehler leichter und schneller zu finden.

Zu Beginn des OSS-Booms (1998) wurde versucht, die Entwicklungsmodelle wissenschaftlich zu erklären. Raymond vergleicht die beiden Entwicklungsmodelle in einem ersten umfassenden Erklärungsansatz als Kathedralen- (klassische, zentralistische Softwareentwicklung) und Basar-Modell (OSS-Entwicklung). Die Kathedrale wird monolithisch und starr erbaut. Der Basar steht als Modell für ein auf den ersten Blick unorganisiertes Durcheinander, welcher auf den zweiten Blick aber eine besondere Leistungsfähigkeit aufgrund des starken Austausches und der Teilnahme der Mitglieder erlangt. Daraus folgt, dass die Wechselwirkung, die gegenseitige Anspornung, die gemeinsame Fehlersuche und das kollektive „Brainstorming“ zu der besonderen Qualität von OSS führen.<sup>13</sup>

Andere haben später das Modell von Raymond kritisch untersucht und auf Schwächen in der Argumentation hingewiesen, dass es auch in OSS-

---

<sup>12</sup> Vgl. Kapitel 2.3

<sup>13</sup> Vgl. Leiteritz (2002), S. 9 f, Raymond (1998)

Entwicklungsteams starke Galionsfiguren gibt und somit auch zentralistische Prozesse (= Kathedralenmodell).<sup>14</sup>

Ein anderer Erklärungsansatz ist das Modell des „Cooking Pot“ von Ghosh. Beim Cooking-Pot Modell schaffen alle Zutaten im Kochtopf einen Wert, der höher ist als die Summe aller Einzelwerte der Zutaten.<sup>15</sup>

### 2.4.3 Unterschiede in den Lizenzen

Die Lizenzierung von Software ist für die proprietäre Softwareindustrie ein wichtiger Bestandteil der Wertschöpfung. Diese Software umfasst folgende Schutzinstrumente:

- *„Verborgener Austausch (Trade Secrecy): Die Software wird als Binärprodukt<sup>16</sup> ausgetauscht. Ein reverse engineering (Schließen auf den ursprünglichen Quellcode) ist dadurch praktisch nicht möglich. Weitere Aspekte sind Verschwiegenheitsvereinbarungen mit Mitarbeitern und Partnern, geltende Rechtsprechung und technische Kopiermechanismen.*
- *Lizenzen: Durch die Lizenzgestaltung kann der Hersteller die Verteilung der Software beschränken. Der Kunde erhält nur ein Nutzungsrecht.*
- *Copyright: Proprietäre Software ist urheberrechtlich geschützt und dadurch nicht 1:1 kopierbar. Vergleichbare Produkte müssen sich technisch unterscheiden.*
- *Patente: Seit Mitte der 80er Jahre ist Software in den USA patentfähig. Mit Hilfe von Patenten können hinter der Software liegende Erfindungen („software-ideas“)<sup>17</sup> geschützt werden.“<sup>18</sup>*

Im Gegensatz dazu, stehen hinter dem Erfolg von OSS offene Lizenzen, die dem Anwender einen hohen Freiheitsgrad lassen. Die wichtigsten OSS-Lizenzen sind im Kapitel 2.6 genauer beschrieben.

---

<sup>14</sup> Vgl. Bezroukov (1999)

<sup>15</sup> Vgl. Ghosh (2000)

<sup>16</sup> In Maschinencode übersetzter Quelltext, für Menschen nicht lesbar

<sup>17</sup> Nicht die Idee ist patentierbar sondern nur der Lösungsweg bzw. Algorithmus

<sup>18</sup> Leiteritz (2002), S. 10

## 2.5 Open Source Software Projekte

### 2.5.1 Klassische Open Source Software

Nachdem im letzten Kapitel auf die Unterschiede zwischen OSS und proprietärer Software eingegangen wurde, werden nun einige bekannte OSS-Projekte aus verschiedenen Bereichen aufgezählt. Eine genauere Betrachtung der Funktionalität wird im Rahmen dieser Master-Thesis nicht durchgeführt.

Die Open Source Bewegung hat zu einer großen Anzahl an Produkten geführt, die sowohl über das Internet, als auch über kommerzielle Distributoren vertrieben werden.

Am Bekanntesten ist das Betriebssystem Linux mit den grafischen Benutzeroberflächen KDE und GNOME. Daneben gibt es weitere offene Betriebssysteme, wie Free/Open BSD.

Ebenso sind Datenbanken als Open Source verfügbar, wie MySQL, PostgreSQL oder SQLite. Auch Office-Produkte stehen als OSS zur Verfügung, wie OpenOffice.org, Gnome Office oder Koffice. Im Grafikbereich sind OSS-Applikationen wie GIMP verfügbar. Auch zur Programmierung der Software sind zahlreiche Werkzeuge wie GNU Compiler, WxWidget oder CASE-Tools<sup>19</sup> verfügbar.

Im Internetbereich sind OSS Produkte bereits seit längerer Zeit Marktführer. So ist die für die Implementierung des Domain Name System und die Zuordnung von Domain Namen zu IP-Adressen zuständige Software BIND<sup>20</sup> ein Open Source Produkt, genauso wie das für die Übermittlung von E-Mails eingesetzte Produkt Sendmail. Lt. neuesten Studien<sup>21</sup> laufen über 70 Prozent aller Internet Server mit der OSS Applikation Apache. Als Browser tut sich Mozilla, der OSS Nachfolger von Netscape Navigator, hervor.

Auch für die Erstellung von Webseiten gibt es zahlreiche OSS-Produkte wie HTML-Editoren, PHP, CGI oder Zope für das Contentmanagement.

---

<sup>19</sup> CASE = Computer Aided Software Engineering

<sup>20</sup> Berkeley Internet Name Domain

Natürlich existieren noch viele weitere Open Source Projekte. Sourceforge.net allein hostet schon über 87 000 aktive Projekte (wovon zugegebenermaßen nicht alle das Prädikat «Open Source» verdienen).<sup>22</sup>

Einzig für Workflowprodukte, Projektmanagementsoftware (MS Projekt) und Flussdiagrammsoftware (MS Visio) gibt es noch keine adäquate Alternative am Open Source Sektor. Es sind jedoch bereits Entwicklungen diesbezüglich in Arbeit, jedoch noch nicht unbedingt empfehlenswert für den Produktiveinsatz in Unternehmen.

## **2.5.2 Kommerzielle Open Source Software**

Neben den reinen Open Source Produkten gibt es auch einige kommerzielle Open Source Produkte für Linux (COLS).

Dabei gilt eine Linux-Distribution nicht als kommerzielles Produkt. Die Kosten für diese Software beziehen sich alleine auf die Zusammenstellung der Distribution, den Support für den Endbenutzer und anderen Dienstleistungen und nicht auf die Kosten für die jeweiligen Programme. Oftmals kann man die idente Distribution kostenlos (abgesehen von ISP Gebühren) über das Internet beziehen.

Ein reines OSS-Produkt ist manchmal auch als kommerzielles OSS-Produkt verfügbar. Durch die Lizenzgestaltung kann bei der Wiederverwendung als proprietärer Bestandteil einer kommerziellen Software eine geringe Lizenzgebühr fällig werden. Ebenfalls gibt es Produkte, welche für den Privatanwender kostenlos verfügbar sind, jedoch für den Unternehmenseinsatz sind Lizenzgebühren zu zahlen.

Als Beispiel für ersteres kann MySQL angeführt werden. Der Sourcecode ist via Internet verfügbar und die Software kann kostenlos eingesetzt werden. Wird MySQL jedoch mit einer proprietären, kommerziellen Software verbunden und verkauft, so muss an die Firma MySQL AP eine Lizenzgebühr in der Höhe von 500,- Euro bezahlt werden.<sup>23</sup>

---

<sup>21</sup> Vgl. DerStandard.at (04.08.2004)

<sup>22</sup> Vgl. Sorgeforce (10.09.2004)

<sup>23</sup> Vgl. MySQL (10.09.2004)

Eine andere Quelle für kommerzielle Linux Software sind Unternehmen, die ihre Windows Software in den letzten Jahren auf Linux portiert haben. Bekannte Unternehmen wie SAP, Oracle, Novell, IBM und viele andere haben diesen Schritt vollzogen, mit dem Ziel, auch auf dem Linuxsektor präsent zu sein.

Diese Software wird jedoch nicht als reine Open Source zur Verfügung gestellt, sondern der Quellcode bleibt weiterhin unter Verschluss. Diese Software wird ebenso wie unter Windows über Lizenzen verkauft. Ebenso ist es schwierig, diese Software als OSS-Projekt zu bezeichnen, da hinter der Software eine Firma steht, die Geld damit verdienen will.

## **2.6 Rechtliche Aspekte**

### **2.6.1 Allgemein**

Freie- und Open Source Software (OSS) darf nicht mit Freeware oder Public Domain-Software verwechselt werden. Freeware erlaubt zwar die kostenlose Benutzung, verlangt aber nicht, dass der Quelltext veröffentlicht wird. Änderungen an der Software sind deshalb weder möglich noch erlaubt.

Der Begriff Public Domain-Software sagt aus, dass die Software „nicht urheberrechtsgeschützt“ ist. Der Autor verzichtet auf alle Rechte an seiner Software und macht sie so zum öffentlichem Eigentum (Public Domain).<sup>24</sup>

Für Open Source gibt es zahlreiche Lizenzen, welche die Anforderungen der Free Software Definition (FSD) oder der Open Source Definition (OSD) vollständig oder auch nur teilweise erfüllen. Jedem, der eine selbst entwickelte Software als OSS vertreiben will, ist freigestellt, ob er sich einer dieser Lizenzen als Modell bedient oder ob er eine eigene Lizenz entwickelt. Insgesamt sind bis heute 52 unterschiedliche Lizenzen entstanden.<sup>25</sup>

---

<sup>24</sup> Vgl. Arpagaus/Seelhofer/Brüngger (2003), S. 82 f

<sup>25</sup> Vgl. DerStandard.at (08.08.2004)

Aus diesem Grund ist bei der Beschaffung von OSS jeweils im Einzelnen zu prüfen, unter welchen Lizenzbedingungen die betreffende Software angeboten wird, was wiederum die Akzeptanz beeinträchtigen kann.

Sowohl die Free Software Foundation als auch die Open Source Initiative (OSI) führen auf ihren jeweiligen Webseiten Lizenzen an und geben bekannt, inwieweit diese mit der FSD und der GNU Public Licence bzw. der OSD kompatibel sind.<sup>26</sup> Die verschiedenen Lizenzen weichen zum Teil erheblich voneinander ab, wobei die GNU General Public License (GPL) und die GNU Lesser General Public Licence (LGPL) die mit Abstand am weitest verbreiteten Lizenzen sind.

Ein zentrales Kriterium zur Beurteilung der verschiedenen Lizenzen ist, ob diese eine Copyleft-Bedingung beinhalten oder nicht. Mit der Copyleft-Bedingung ist es den Nutzern der Software erlaubt, Teile der freien Software in andere Software zu übernehmen und diese als proprietäre Software zu vertreiben.

Lizenzen, die ein Copyleft beinhalten, verlangen demgegenüber, dass jede Software, die unter Modifikation der freien Software oder unter der Übernahme von Teilen derselben erstellt wurde, ebenfalls wieder als freie Software mit offenem Quellcode zur Verfügung gestellt wird.<sup>27</sup>

Neben den allgemeinen Aspekten der Lizenzmodelle gibt es auch bei den OSS-Produkten noch folgende rechtliche Aspekte zu klären:<sup>28</sup>

- Urheberrecht: Der Urheber bestimmt die Nutzung seiner Software. Der Lizenznehmer hat jedoch weitgehende Rechte im Zusammenhang mit der Vervielfältigung, dem Vertrieb und der Änderung des Quellcodes.
- Patentrecht: Ist kontrovers zum OSS-Gedanken und wird in der Europäischen Union heftig diskutiert. Mittels Patentrecht kann die Verwendung von Softwarekomponenten untersagt werden.

---

<sup>26</sup> Für eine Liste von Free Software Lizenzen Vgl. Free Software Foundation (02.09.2004)

<sup>27</sup> Vgl. Widmer (2003), S. 12

<sup>28</sup> Vgl. Widmer (2003), S. 20, Arpagaus/Seelhofer/Brüngger (2003), S. 82

- Markenrecht: Das Markenrecht ist mit GPL oder LGPL nicht vereinbar, da dies den freien Vertrieb der Software einschränken würde.
- Wettbewerbsrecht: Von Interesse ist nur das Prinzip der kostenlosen Lizenzierung von OSS bei der Weiterverbreitung in gleicher oder veränderter Form.
- Schutzberechtigte: Darunter versteht man die Mitarbeiter an einem OSS-Projekt und deren Schutz auf Grund der vier zuvor genannten rechtlichen Aspekte.
- Vertragsrecht: Es fehlen oftmals lokalisierte Versionen der Lizenzmodelle, meist sind sie nur in Englisch verfügbar. Da stellt sich die Frage nach der Rechtsgültigkeit.
- Haftungsrecht: Es muss festgestellt werden, wer für auftretende Mängel haftbar gemacht werden kann.

Eine genauere Bearbeitung dieser oben genannten Punkte ist nicht für diese Master-Thesis vorgesehen und wird somit nicht ausgeführt. Als Nachschlagewerk werden einschlägige rechtswissenschaftliche Artikel empfohlen.

## **2.6.2 OSS Lizenzmodelle**

### 2.6.2.1 GNU General Public License (GPL)

Dieses Lizenzmodell geht auf den amerikanischen Softwarespezialisten Richard Stallman, der das GNU Projekt schuf, zurück. Die GPL ist der bekannteste Vertreter der OSS-Lizenzmodelle. Laut Definition der „Freien Software“ (FSD) gibt es folgende Freiheiten:<sup>29</sup>

1. Freiheit zur Nutzung eines Programms zu beliebigen Zwecken
2. Freiheit zum Studium der Funktionsweise eines Programms und zur Anpassung an die eigenen Bedürfnisse
3. Freiheit zum Weitervertrieb von Programmen

---

<sup>29</sup> Vgl. Widmer (2003), S. 10

4. Freiheit zur Modifikation und Verbesserung von Programmen und zur Veröffentlichung solcher Modifikationen und Verbesserungen, um diese allgemein nutzbar zumachen

Diese Freiheiten werden durch das Konzept des so genannten „Copyleft“ ergänzt.

*„Copyleft is a general method for making a program free software and requiring all modified and extended versions of the program to be free software as well.“<sup>30</sup>*

Copyleft soll die Andersartigkeit gegenüber Copyright (= Urheberrecht) verdeutlichen.

Die GPL beinhaltet neben der freien Verfügbarkeit des Quellcodes und dem Recht diesen zu ändern auch, dass Weiterentwicklungen oder Änderungen wieder frei verfügbar gemacht werden müssen. Somit hat dieses Lizenzmodell einen „viralen“ Charakter indem es sich immer weiter verbreitet.

Somit ist die Verwendung von GPL geschützter Software für kommerzielle Softwareunternehmen unattraktiv, da immer der komplette Quellcode frei gegeben werden muss. Diese Lizenz bereitete manchen Entwicklern Unbehagen, so dass sich mit der Zeit weitere Lizenzmodelle entwickelt haben.<sup>31</sup>

#### 2.6.2.2 GNU Lesser General Public License (LGPL)

Mit der LGPL wurde eine alternative Lizenz entwickelt, welche den (auch kommerziellen) Vertrieb von Software unter einer anderen Lizenz erlaubt, sofern diese die Funktionalität einer unter LGPL stehenden Bibliothek nur benutzt (und diese nicht erweitert). Auch der Name des Entwicklers muss im Sourcecode verbleiben und darf nicht gelöscht werden.

Mit der GPL wäre das nicht möglich.<sup>32</sup>

---

<sup>30</sup> Free Software Foundation (15.06.2004)

<sup>31</sup> Vgl. Leiteritz (2002), S. 11 , Arpagaus/Seelhofer/Brüngger (2003), S. 83

<sup>32</sup> Vgl. Arpagaus/Seelhofer/Brüngger (2003), S. 83

### 2.6.2.3 BSD-Lizenz

Die BSD-Lizenz ist viel toleranter als die GPL. Sie verlangt nur, dass die originalen Copyright-Notizen, die Bedingungen und der Disclaimer in den Quellcode-Dateien belassen werden.

Mit dieser Lizenz darf OSS auch in proprietäre Software eingebunden und kommerziell genutzt werden, wenn die genannten Angaben in gedruckter oder elektronischer Form mitgeliefert werden.

Einzige Einschränkung ist, die Namen der Autoren dürfen nicht für Werbezwecke verwendet werden. Somit haben viele Produkte BSD-konformen Quellcode eingebaut, sogar Microsoft in Windows.<sup>33</sup>

### 2.6.2.4 Mozilla Public License (MPL)

Die MPL entstand durch die Freigabe des Quellcodes vom Netscape Communicator für die OS-Gemeinde. Es ist eine mit GPL verwandte Lizenz, welche von der Netscape Rechtsabteilung entwickelt wurde. Auch bei der MPL müssen Modifikationen wieder unter MPL und im Quellcode verfügbar gemacht werden.<sup>34</sup>

### 2.6.2.5 Duale Lizenzmodelle

Wie schon an Hand des Beispiels MySQL<sup>35</sup> erwähnt, gibt es Open Source Produkte mit verschiedenen Lizenzen. Somit kann eine Open Source Software kommerziell angeboten werden.

Aus dem Beispiel folgt, dass MySQL grundsätzlich unter GPL angeboten wird. Jedoch sind auch kommerzielle Lizenzen verfügbar, mit denen MySQL in proprietären Softwareprodukten eingesetzt und verkauft werden darf.<sup>36</sup>

---

<sup>33</sup> Vgl. Arpagaus/Seelhofer/Brüngger (2003), S. 84

<sup>34</sup> Vgl. Arpagaus/Seelhofer/Brüngger (2003), S. 84

<sup>35</sup> Vgl. Kapitel 2.5.2

<sup>36</sup> Vgl. Arpagaus/Seelhofer/Brüngger (2003), S. 84

### 2.6.3 Unterschiede zwischen den Lizenzmodellen

Die offenen Lizenzen haben zum Erfolg von OSS beigetragen, da sie dem Anwender einen hohen Freiheitsgrad lassen. Die folgende Tabelle soll einen Vergleich der jeweiligen Eigenschaften bieten.<sup>37</sup>

	Kostenlos	Freie Verteilung	Uneingeschränkter Gebrauch	Quellcode vorhanden	Quellcode modifizierbar	Derivate müssen frei sein	Keine Vermischung mit proprietärer Software
Proprietär	-	-	-	-	-	-	-
Shareware	?	Ja	-	-	-	-	-
Freeware	Ja	Ja	Ja	-	-	-	-
BSD	Ja	Ja	Ja	Ja	Ja	-	-
LGPL	Ja	Ja	Ja	Ja	Ja	Ja	-
GPL	Ja	Ja	Ja	Ja	Ja	Ja	Ja

Tabelle 1: Vergleich der Eigenschaften wichtiger OSS-Software-Lizenzen

(aus: Leiteritz (2002), S. 11; Opensource.org (09.04.2004))

## 2.7 OSS Organisationen

### 2.7.1 GNU und die Free Software Foundation (FSF)

Das GNU Projekt wurde im Jahre 1984 von Richard Stallman ins Leben gerufen, um ein Unix-ähnliches Betriebssystem zu entwickeln, welches frei verfügbar ist. Ein Jahr später gründete Stallman die Free Software Foundation (FSF). Diese Foundation kümmert sich um die Geldbeschaffung für das GNU-Projekt und unterstützt das Projekt mit einer Organisation.

Das Wort „frei“ in „freier Software“ wird von der FSF mit „Freiheit“ und nicht mit „kostenloser Verfügbarkeit“ interpretiert. Damit ist gemeint, dass freie Software sowohl beliebig verteilt und modifiziert werden darf, als auch weiterhin frei und im Sourcecode verfügbar bleiben muss und nicht in ein kommerzielles Paket übergehen darf.<sup>38</sup>

<sup>37</sup> Vgl. Leiteritz (2002), S. 11, Arpagaus/Seelhofer/Brüngger (2003), S. 83, Opensource (10.09.2004)

<sup>38</sup> Vgl. Arpagaus/Seelhofer/Brüngger (2003), S. 74

### **2.7.2 Open Source Initiative (OSI)**

Die Open Source Initiative (OSI) wurde im Jahre 1998 als non-profit Organisation mit dem Ziel gegründet, die Entwicklung von Open Source Software zu fördern. Auslösendes Ereignis war Netscapes Ankündigung, den Quellcode ihrer Browser-Software (Netscape Communicator) freizugeben.

Die OSI stellt einen Zertifizierungsprozess zur Verfügung, um Open Source Software eindeutig als solche identifizieren zu können.

OSI versteht unter Open Source Software frei verfügbaren Quellcode. Einzig die Verfügbarkeit zählt, somit ist es auch möglich Open Source Software in kommerzielle Software-Pakete einzubinden oder modifizierte Versionen direkt zu verkaufen.

Laut OSI ist der Entwicklungsprozess von Open Source Software demjenigen von proprietärer Software überlegen. Die Mitglieder der OSI sind davon überzeugt, dass das „Vielaugenprinzip“ der Open Source Gemeinschaft eine bessere und sicherere Software hervorbringt.<sup>39</sup>

### **2.7.3 Apache Software Foundation (ASF)**

Die ASF ist 1995 aus dem ersten Webserver Projekt hervorgegangen. Es ist eine juristische Körperschaft zur Unterstützung in Organisation, Rechtsfragen und Geldbeschaffung.

Seit April 1996 ist der auf diversen Plattformen verfügbare Apache erhältlich. Heutzutage ist er der beliebteste Webserver der Welt und sein Marktanteil stieg bis September 2004 kontinuierlich auf ca. 70% an. Außer dem Webserver-Projekt entwickelt die ASF noch weitere Open Source Projekte.<sup>40</sup>

### **2.7.4 Open Source Development Network (OSDN)**

Das Open Source Development Network (OSDN) wurde im Jahr 2000 von dem Unternehmen VA Software gegründet. Das OSDN ist ein Verbund verschiedener Plattformen mit dem Ziel, die Entwicklung von Open Source

---

<sup>39</sup> Vgl. Arpagaus/Seelhofer/Brüngger (2003), S. 74

Software zu beschleunigen und sowohl Einzelentwickler als auch Unternehmen an die Idee der gemeinsamen OSS-Entwicklung heranzuführen. Hervorgegangen ist das OSDN aus Andover.net, einem Netzwerk führender Open Source Internetangebote wie Slashdot.org und Linux.com. Heutzutage kennt man OSDN eher unter der Plattform „Sourceforge.net“, einer Plattform zur Unterstützung des Entwicklungsprozesses von Software mit einer großen Zahl von Entwicklern. Sourceforge.net, welche von VA Software gesponsert wird, stellt der Open Source Gemeinschaft kostenlos Speicherplatz zur Verfügung und ist dadurch die weltweit größte Entwicklungsplattform (hostet über 87.000 Projekte) geworden.<sup>41</sup>

## **2.8 Distributionen**

Die Positionen der Hersteller proprietärer Software wie Microsoft, IBM, HP, Oracle oder Sun zu Open Source Software werden in dieser Thesis nicht angeführt, da diese für das Ergebnis dieser Thesis nicht relevant sind. Für detailliertere Informationen werden einschlägige Fachzeitschriften oder die jeweiligen Firmenhomepages empfohlen.

Eine wichtige Aufgabe in der Verbreitung von Open Source Software haben insbesondere die Distributionen. Nicht jeder Anwender verfügt über einen ausreichend schnellen Internetanschluss, um sich die umfangreichen Softwarepakete herunter zu laden. Insbesondere für Neueinsteiger in die Linuxumgebung bieten Distributionen Vorteile, da sie teils umfangreiche Benutzerhandbücher beinhalten, was bei proprietärer Software oftmals nicht mehr der Fall ist (z.B: Microsoft Windows)

Es gibt eine Vielzahl von Distributionen, jedoch werden im Folgenden nur die beiden Wichtigsten für Unternehmen vorgestellt.

---

<sup>40</sup> Vgl. Arpagaus/Seelhofer/Brüngger (2003), S. 74 f

<sup>41</sup> Vgl. Arpagaus/Seelhofer/Brüngger (2003), S. 75, Sourceforge (10.09.2004)

Wenn man den offiziellen Weg gehen will und weitgehend auf zertifizierte Software setzt, dann hat man als Unternehmen zumeist nur die Wahl zwischen Red Hat Professional System oder Novell SUSE Enterprise Server.

Beide Anbieter bieten lange, unterstützte Releasezyklen, in denen die Distribution stabil bleibt und nur Bugfixe angeboten werden. So soll die Investition der Kunden geschützt und auch deren Aufwand minimiert werden. In diese Lösungen ist auch ein Update-Mechanismus eingebaut, der von einer zentralen Stelle die Bugfixe ins System einspielt. Diesen Distributionen gemeinsam ist, dass sie in ihren Paketen auch die Wartung und den Support inkludieren. Dies gilt zumeist ein Jahr, danach muss man neue Verträge abschließen.

Die Distributionen der beiden Anbieter sind nicht gerade günstig, im Vergleich zu den Distributionen für den Privatanwender. Wenn der Systemadministrator eines Unternehmens kompetente Linuxkenntnisse aufweist, kann auch auf eine billigere Lösung anderer Hersteller wie Mandrake, Debian oder SoL zurückgegriffen werden.<sup>42</sup>

---

<sup>42</sup> Vgl. WCM – Computerzeitung für Österreich (2004), S. 32 f

## 3. Ist-Situation

„Microsoft wurde bereits beim Webserver  
von einem Open-Source-Produkt  
vernichtend geschlagen“

(Oracle-Chef Larry Ellison beim Vergleich  
des Open-Source-Webservers Apache mit dem  
Microsoft-eigenen Internet Information Server)

### 3.1 Verbreitung

Das Angebot an verfügbarer und qualitativ hochwertiger Open Source Software (OSS) hat in den vergangenen Jahren deutlich zugenommen. Es existieren weltweit mehrere zehntausend Projekte, die OSS für die unterschiedlichsten Anwendungsbereiche und für unterschiedlichste Zwecke entwickeln.

Federführend sind dabei die Distributionen wie Suse oder Red Hat, die verschiedenste Produkte bündeln und als Gesamtpaket verkaufen oder via Internet verfügbar machen. Darüber hinaus bieten diese Distributoren für die jeweiligen Programme Benutzerhandbücher, Support und Wartungsdienste gegen Entgelt an.

Während OSS im Server- und Internetbereich bereits weit verbreitet ist, so ist OSS am Desktop-Markt fast nicht vertreten.

Laut einer Studie der IT-Berater Cambridge Technology Partners über den Einsatz, die Vor- und Nachteile von Open Source Software in 55 der 450 größten Schweizer Unternehmen ergab, dass bereits 70 Prozent der befragten Firmen OSS in irgendeiner Form einsetzen. Umgelegt auf kommerzielle Software entspräche dies einem Lizenzvolumen von 500 Millionen Schweizer Franken (ca. 335 Millionen Euro).<sup>43</sup>

---

<sup>43</sup> Vgl. Computerworld (2003), S. 3

Einen umfassenden Überblick über die Verbreitung von OSS-Produkten und die für deren Beschaffung, Nutzung und Entwicklung maßgeblichen Gründe in Deutschland, England und Schweden geben mehrere Berlecon Studien.<sup>44</sup>

### **3.2 Empfehlenswerte Einsatzgebiete**

Diverse Studien erwähnen Einsatzgebiete, in denen der Einsatz von OSS empfohlen wird:<sup>45</sup>

- Internetbereich: OSS eignet sich wegen des offenen Internetstandards und des großen Softwareangebots gut, wenn ein Unternehmensnetzwerk an das Internet angebunden werden soll (z. B. E-Mail, Webserver).
- Intranetbereich<sup>46</sup>: OSS eignet sich aufgrund der standardisierten Protokolle und der Kompatibilität mit den Internet-Technologien besonders gut.
- Virtual Private Network: (VPN): OSS eignet sich aufgrund der erhöhten Sicherheit, um mittels VPNs Niederlassungen eines Unternehmens abgesichert über das Internet zu verbinden.
- Heterogenes Netz: Durch die vielfältigen Protokolle und offenen Schnittstellen eignet sich OSS sehr gut, wenn unterschiedliche Computersysteme und Netzwerke verbunden werden.
- Datenarchivierung: OSS eignet sich besonders gut für langfristige Dokumentarchivierung, aufgrund der vielen herstellerunabhängig unterstützten Datenformate.

---

<sup>44</sup> Näheres siehe: Berlecon Research (2002)

<sup>45</sup> Vgl. BMWi/Innominate AG (2001), S. 44 f

<sup>46</sup> Ein Intranet dient zur internen Vernetzung in einem Unternehmen.

## 3.3 Marktsituation

### 3.3.1 Allgemeines

Für die Beschreibung des Marktes der Informationstechnologie im Allgemeinen hat Hoch das folgende Modell entwickelt:

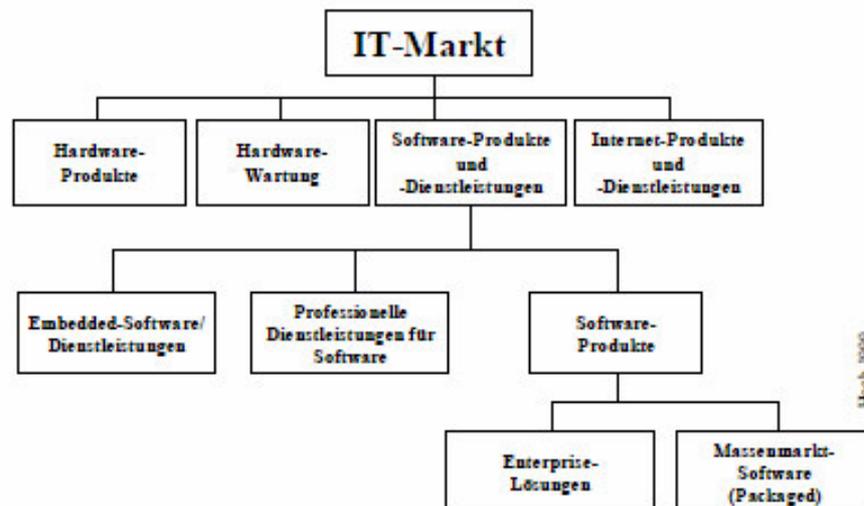


Abbildung 3: Struktur des IT-Marktes (aus Leiteritz (2002), S. 17)

OSS hat sich in vielen, aber nicht allen Segmenten des IT-Marktes entwickelt. So ist das Hardware-Segment nicht relevant, da es sich bei OSS per Definition um Software handelt.

Es gab in den letzten Jahren zwar Ideen, das Prinzip von offenen Quellen auch auf Hardware, z. B. auf CPUs zu übertragen, aber diese Ansätze sind bisher nie über das theoretische Stadium hinausgekommen und haben deshalb keine Relevanz im Sinne von in die Praxis umgesetzten analysierbaren Geschäftsmodellen.

Für die weitere Untersuchung der relevanten OSS-Teilmärkte wird dieser in den Produktmarkt, den Dienstleistungsmarkt und den Internetmarkt unterteilt.

Der Software-Produktmarkt unterscheidet sich generell in einigen Eigenschaften von anderen Märkten. Hoch hat eine Reihe von Merkmalen definiert:

- *„Niedrige Eintrittsbarrieren: Die Markteintrittsbarrieren sind im Software-Markt gering. Softwareentwicklung ist wissensorientiert und deshalb leicht kopierbar.*
- *Niedrige Investitionen: Im Softwaremarkt sind keine Fabriken, keine Grundstücke und keine Maschinen notwendig. Für die ersten Schritte reichen Büroausstattung, Internetzugang, Hard- und Software und Softwareentwickler.*
- *Schnelle Innovationszyklen: Da die Eintrittsbarrieren niedrig sind und wenig Kapital benötigt wird, streben viele Anbieter auf den Markt. Diese setzen auf Innovationen und treiben bestehende Unternehmen wiederum zu neuen Innovationen.*
- *Niedrige Grenzkosten: Der erste Datenträger einer Software kann Millionen von Euro kosten, jeder weitere nur noch Cents. In der Softwareindustrie existieren praktisch keine Produktionskosten. Der gesamte Aufwand fällt in der Entwicklung und der Vermarktung der Software an. Deshalb wird Software in möglichst großen Stückzahlen für möglichst große Märkte produziert.*
- *Kampf um die Marktführerschaft: Softwareunternehmen profitieren überdurchschnittlich stark von hohen Marktanteilen. Dieses Phänomen ist unter dem Namen „Law of Increasing Returns“ bekannt.<sup>47</sup> Produkte, die hohe Marktanteile haben und weiter Marktanteile gewinnen, wachsen wiederum überdurchschnittlich stark. Produkte mit fallenden Marktanteilen verlieren besonders viele Marktanteile. Der Grund für dieses Phänomen ist, dass der Nutzen für die Anwender bei hohen Marktanteilen größer wird (niedrigere Schulungskosten, Gewöhnung, einfachere Bedienung, Bevorzugung erfolgreicher Software, höhere Wechselkosten) und dieses wiederum positive Feedback weiteres Wachstum erzeugt.“<sup>48</sup>*

---

<sup>47</sup> Vgl. Zerdick et. al (1999), S. 157

<sup>48</sup> Leiteritz (2002), S. 17

### 3.3.2 Produktmarkt

#### 3.3.2.1 Grundlegendes

Am Beginn dieses Abschnittes wird ein Überblick über den Einfluss von OSS in den verschiedenen Märkten gegeben. Dabei steht „+“ für hohe Relevanz, „o“ für mittlere Relevanz und „-“ für niedrige Relevanz:

	<b>Server</b>	<b>Client</b>	<b>Embedded</b>
<b>Betriebssystem</b>	Linux-Distribution +	Linux-Distribution o	Linux-Distribution o
<b>Applikationen</b>	Server-Applikationen +	Desktop-Applikationen +	Embedded Applikationen o
<b>Appliances</b>	Server Appliances o	Thin Client / Set Top Box o	PDA, MP3-Player, Smartphone o

Tabelle 2: OSS-Produktmarktsegmente und deren Relevanz im kommerziellen Bereich

(vgl: Leiteritz (2002), S. 18)

Die Übersicht ist nach den Zielplattformen (Server, Client oder Embedded) und in den Umfang des Produktes (Betriebssystem, Applikation oder Appliances<sup>49</sup>) unterteilt. „Relevanz“ steht in der Abbildung für die Einschätzung der kommerziellen Bedeutung von Open Source/Linux.<sup>50</sup>

---

<sup>49</sup> Appliances sind Betriebssystem/Applikationen/Hardware-Kombinationen (z. B. vorinstallierte E-Mail-Server).

<sup>50</sup> Vgl. Leiteritz (2002), S. 18

Eine Studie von Soron über die Entwicklung des Open-Source-Marktes in Deutschland hat zu folgendem Ergebnis geführt (in Millionen Euro):<sup>51</sup>

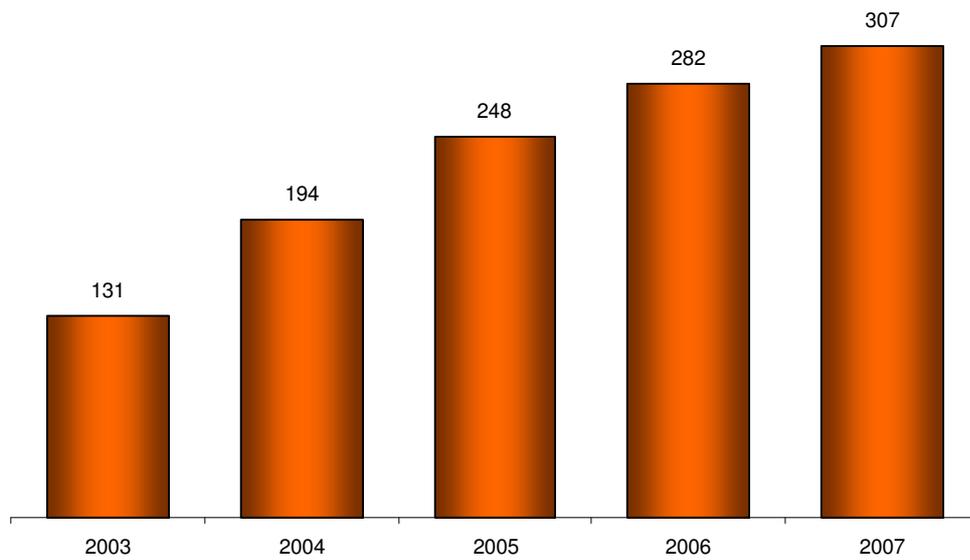


Abbildung 4: Gesamter Open Source Unternehmensmarkt 2003 (aus: Computerworld Magazin (2003), S.39)

Dieser Trend ist auch auf die anderen deutschsprachigen Länder übertragbar.

---

<sup>51</sup> Vgl. Computerworld-Magazin (2003), S. 39

### 3.3.2.1 Der Markt für Betriebssysteme

Der Markt kann im Allgemeinen in die drei Zielplattformen Server, Client und Embedded eingeteilt werden.

Als Erstes wird der Server-Markt betrachtet. Wie die folgende Abbildung der installierten Server im Jahre 2002 in Deutschland auf Intel-kompatiblen Rechnern (x86 Plattform) zeigt, dominiert Microsoft mit dem Windows Betriebssystem.

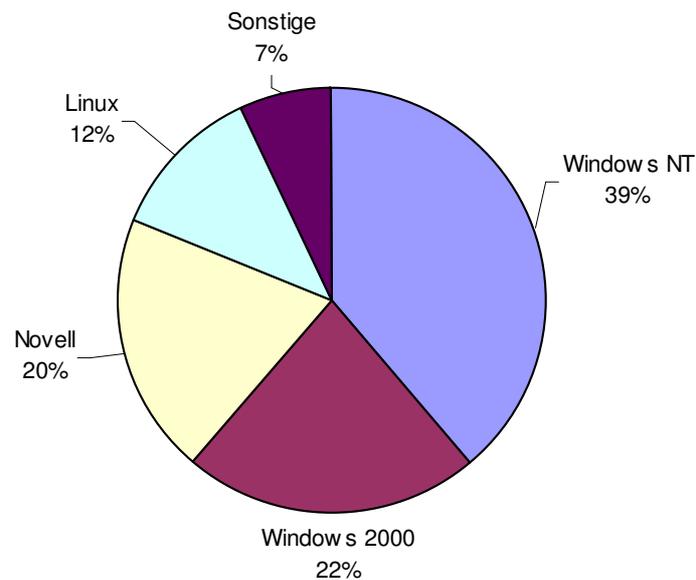


Abbildung 5: Installierte Server 2002 auf x86-Basis (aus: Wendt (2003), Folie 3)

Die Aussage dieser Abbildung ist, dass der OSS-Anteil sehr gering ist und das ca. 60 Prozent der Server migriert werden könnten.<sup>52</sup>

Im Gegensatz zur weitläufigen Meinung, dass Linux auf Kosten von Microsoft Windows wächst bewahrheitet sich nicht. Im Gegenteil, Linux nimmt dem etablierten UNIX Betriebssystem, z. B. Solaris von Sun und AIX von IBM, Marktanteile weg. In der obigen Abbildung ist Unix nicht enthalten, da Unix auf eine andere Plattform aufsetzt.

Andere OSS Betriebssysteme sind FreeBSD und OpenBSD, die vor allem im Bereich Sicherheit und Internetserver eingesetzt werden. Die Gartner

---

<sup>52</sup> Vgl. Wendt (2003), Folie 3

Group schätzt, dass der weltweite Linux Server-Markt von 2001 bis 2005 von \$2,5 Milliarden auf \$9,1 Milliarden wächst. Das Marktforschungsunternehmen Techconsult beurteilte die Branchenverbreitung von Linux bis 2003: Laut dieser Untersuchung haben im kommerziellen Bereich die Energie- und Wasserversorger und die Telekommunikationsunternehmen mit 25 % den höchsten Linux-Einsatzgrad vorzuweisen.

Als Zweites betrachten wir den Client Markt. Der Client Markt wird von Microsoft Windows beherrscht. Die Konkurrenzsysteme Apple und OSS spielen nur eine untergeordnete Rolle, so dass Linux derzeit nur einen Marktanteil von ca. 4% ausweisen kann.<sup>53</sup>

Wie die folgende Abbildung der installierten Clients im Jahre 2002 in Deutschland auf Intel-kompatiblen Rechnern (x86 Plattform) zeigt, dominiert Microsoft mit dem Windows Betriebssystem.

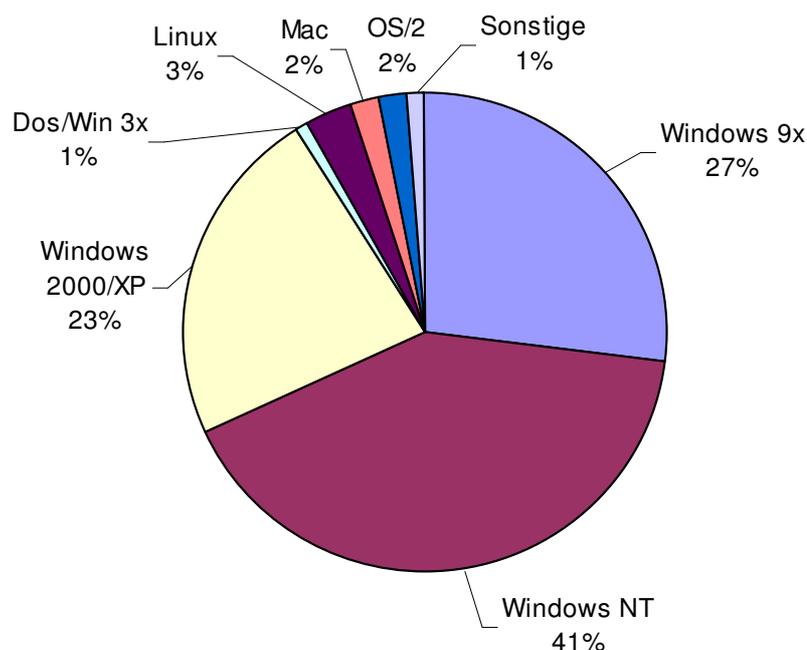


Abbildung 6: Installierte Clients 2002 auf x86- Basis (aus: Wendt (2003), Folie 3)

Die Aussage dieser Abbildung ist, dass der OSS-Anteil noch geringer als beim Server ist und über 70 Prozent der Clients migriert könnten werden.<sup>54</sup>

<sup>53</sup> Vgl. Leiteritz (2002), S. 19 f

<sup>54</sup> Vgl. Wendt (2003), Folie 3

Andere Betriebssysteme spielen auf dem Desktop keine nennenswerte Rolle.

Als Letztes erfolgt die Marktbetrachtung für Embedded Systeme. Embedded-Systeme sind Rechnersysteme, die für den Anwender nicht auf den ersten Blick erkennbar sind. Sie werden vor allem in der Automobilindustrie, Automatisierungsindustrie und in kleinen tragbaren Geräten verwendet. Weit verbreitet ist MS Windows Pocket PC, aber Linux holt auf.

OSS ist in der Embedded-Industrie erst seit einigen Jahren vertreten. Allerdings wird OSS aufgrund der Flexibilität, Offenheit, den Preisvorteilen und der Stabilität eine große Zukunft prophezeit, aber es existieren noch Schwächen z. B. im Echtzeitbereich.<sup>55</sup>

Wie man an folgender Abbildung erkennen kann, entwickelt sich der Weltmarkt für Embedded-Linux Systeme bis 2005, laut Marktbeobachtern, mit einer durchschnittlichen Jahreswachstumsrate von ca. 61 Prozent (in Millionen Dollar).<sup>56</sup>

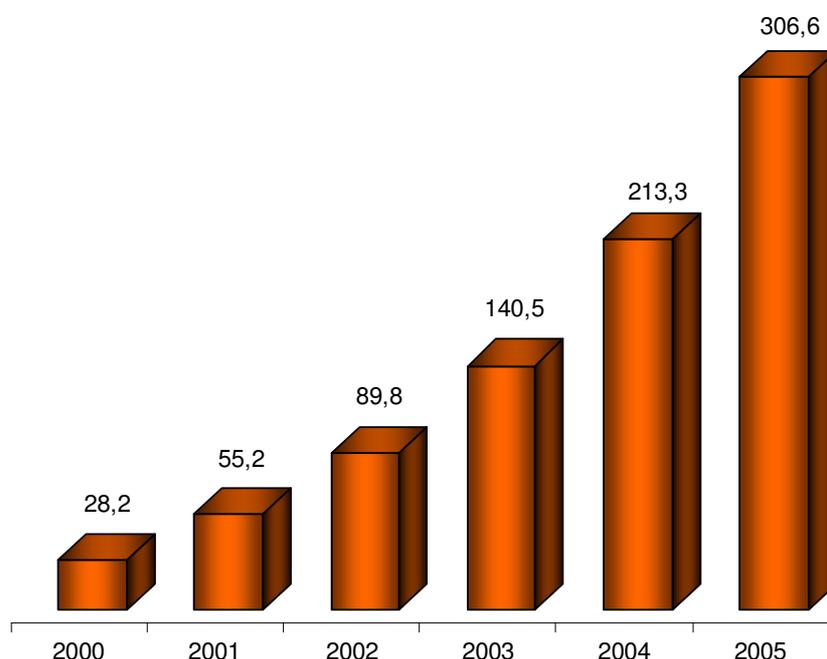


Abbildung 7: Entwicklung des Weltmarkts für Embedded Linux, Software-Entwicklungstools und Dienste (aus: Computerworld Magazin (2003), S. 6)

<sup>55</sup> Vgl. Leiteritz (2002), S. 19 f

<sup>56</sup> Vgl. Computerworld-Magazin (2003), S. 6

### 3.3.2.2 Der Markt für Applikationen

Auch hier wird zwischen Server, Client und Embedded unterschieden.

Für den Server-Bereich werden zentrale Dienste benötigt, wie Internet/Intranet-Server (E-Mail, Webserver), betriebswirtschaftliche Standardsoftware (Buchhaltung), System-Management-Software (Verwaltung von Computersystemen), Datenbank-Software oder Workflow-Software (Steuerung von Arbeitsprozessen)

Es existieren unterschiedlichste Produkte von einer großen Anzahl von Anbietern. OSS hat den Schwerpunkt im Bereich IT-Infrastrukturlösungen wie Internet/Intranet-, Server- und Sicherheitssoftware sowie Datenbankserver. Es existieren zwar Lösungen für die anderen erwähnten Beispiele, jedoch befinden sie sich meist erst im Beta-Stadium und werden noch nicht als Produktionsumgebung empfohlen.

Auch der Client-Bereich ist sehr fragmentiert. Sowie im Server Bereich existiert auch hier eine Vielzahl von Anbietern. Beispiele für Applikationen sind:

- Office Software (Textverarbeitung, Tabellenkalkulation, Präsentation)
- Produktivitätstools (Projektverwaltung)
- Management Information Systeme
- Spiele
- Bild- und Videobearbeitung (Digitalfotografie, Videoschnitt)

Es existieren zwar zahlreiche Produkte, jedoch weist OSS nur einen geringen Marktanteil auf und spielt noch keine bedeutende Rolle. Da der Datenaustausch mit weit verbreiteten kommerziellen Produkten nach wie vor schwierig ist, finden OSS Produkte eher bei technisch versierten Anwendern Verbreitung.

Der Embedded-Markt ist im Gegensatz zu Server- und Clientmarkt kein Massenmarkt. Typische Kunden sind Industrieunternehmen, die eigene Lösungen entwickeln. Es gibt eine überschaubare Anzahl von Anbietern wie z. B. Windriver, Lynx, Motorola und QNX, die neben dem eigentlichen

Betriebssystem auch Applikationen wie z. B. Entwicklungsumgebungen, Managementsoftware und ähnliches anbieten. OSS-Anbieter sind z. B. Lineo (USA) und Sysgo (Deutschland).<sup>57</sup>

### 3.3.2.3 Der Markt für Appliances

Appliances sind Kombinationen von Hardware, Software und Betriebssystem. Je nach Einsatzzweck sind Server- oder Clientgeräte erhältlich. Diese Geräte integrieren mehrere Funktionen in einem Gerät und werden als einfach zu bedienende Komplettlösung verkauft.

Im Aufbau sind Appliances Standardgeräte, welche aus Standard Hardware (PC) und Standard Software (Windows, Linux) bestehen. Diese Appliances richten sich an Kunden, die einen konkreten Lösungsbedarf haben, welche durch die Appliances billiger zu lösen sind, als Individuallösungen.

Durch die Vorinstallation und Vorkonfiguration sind diese Geräte bequem in der Handhabung und können schnell eingesetzt werden.

Beispiele für Appliances im Server-Bereich sind E-Mail Appliances (organisieren Austausch von E-Mails), Firewall Appliances (Internet Sicherheitssystem) und Kommunikationsserver (Internetzugang, E-Mail Austausch, Firewall-Schutz, Fax Dienste und Proxy in Einem).

Beispiele für Client Appliances sind Thin Clients (einfache Arbeitsplatzrechner) und Set-top-Boxen (Heimelektronikgeräte für Anbindung an den Fernseher).<sup>58</sup>

---

<sup>57</sup> Vgl. Leiteritz (2002), S. 21 f

<sup>58</sup> Vgl. Leiteritz (2002), S. 23 f

### 3.3.3 Dienstleistungsmarkt

Im Dienstleistungsmarkt kann man zwischen „Professional Services“ und „Enterprise Solutions“ unterscheiden. Der Unterschied zwischen den beiden Segmenten liegt im Grad der „Produktisierung“ und der Verkaufszahl. Professional Services sind hochindividuelle Dienstleistungen, die nur einmal bzw. wenige Male (1 – 9 Mal) in einer bestimmten Form für einen Kunden erbracht werden. Enterprise Solutions hingegen sind Dienstleistungen mit Produktcharakter. Sie sind zu einem gewissen Grad standardisiert und können für eine größere Anzahl (mehr als 10 Mal) an Kunden verkauft werden.

Sowie der Produktmarkt, kann auch der Dienstleistungsmarkt in verschiedenen Segmenten eingeteilt werden. So können Dienstleistungen nach der technischen Ebene (z.B: Betriebssystem, Server, Client, Embedded-System), den Wertschöpfungsabschnitten (z.B: Beratung, Studien, Implementierung, Support) und der Zielgruppe (z.B: KMU, Enterprise, Finanzunternehmen) unterschieden werden.

Der Dienstleistungsmarkt hat besondere Eigenschaften, welche ihn vom Produktmarkt unterscheiden:<sup>59</sup>

- Hohe und konstante Grenzkosten: Im Gegensatz zu den Produkt-Geschäftsmodellen fallen bei Dienstleistungen bei jedem Erstellungsvorgang nahezu identische variable Kosten an, da jedes Mal Personen involviert sind. McKinsey beschreibt, dass diese Infrastrukturkosten bei Dienstleistungsunternehmen mehr als viermal höher sind als bei Software-Produktunternehmen. Dies ist der Grund, warum das „Law of Increasing Returns“, also die zunehmende Profitabilität bei steigendem Produktabsatz, bei Dienstleistungsunternehmen nicht gilt.
- Niedrige Fixkosten: Die Kosten, um ein Dienstleistungsunternehmen zu gründen, sind niedriger als bei einem Produkt-Geschäftsmodell. Bei der Produktentwicklung muss eine Entwicklungszeit ohne Umsätze finanziert werden. Für ein Dienstleistungsmodell reichen wenige Mitarbeiter und eine einfache Büroinfrastruktur aus.

- Hoher Wettbewerb/Fragmentierung: Niedrige Fixkosten machen den Dienstleistungsmarkt sehr wettbewerbsintensiv. Da Dienstleistungsunternehmen aber keine Möglichkeit haben, überproportional von Markterfolgen durch das „Law of Increasing Returns“ zu profitieren, ist der Dienstleistungsmarkt geographisch fragmentiert (jedes Jahr werden in den USA mehr als 10.000 IT-Dienstleistungsunternehmen neu gegründet) und die Anbieter sind relativ klein.
- Optimierung der Auslastungsquote: Software-Unternehmen können Erfolg anhand der Marktanteile ihrer Produkte messen. Dienstleistungsunternehmen müssen Erfolg an der (internen) Größe des Auslastungsgrads messen. Das Personal verursacht Fixkosten, egal ob Dienstleistungsprojekte verkauft werden oder nicht. Nur wenn die Umsätze der verkauften Projekte höher sind als die Personalkosten und die sonstigen Kosten, werden Gewinne erzielt.
- Verkauf vor Fertigstellung: Software-Produkte werden aufgrund von definier- und vergleichbaren Eigenschaften verkauft. Dienstleistungsprojekte werden verkauft, ohne dass der Kunde sicher sein kann, die versprochene Leistung auch zu erhalten. Deshalb sind Vertrauen und eine enge Kundenbeziehung notwendig.

IDC beschreibt die Größe des Marktes für Linux-Dienstleistungen wie folgt:<sup>60</sup>

	<b>2000</b>	<b>2001</b>	<b>2002</b>	<b>2003</b>	<b>2004</b>
Umsatz in Mio. \$	28,3	56,6	105,0	177,0	285,0
Wachstum in %		97,9%	87,5%	68,6%	61,0%

Tabelle 3: Der Markt für Linux-Dienstleistungen (aus: Leiteritz (2002), S. 25)

<sup>59</sup> Vgl. Leiteritz (2002), S. 24 f

<sup>60</sup> Vgl. Leiteritz (2002), S. 25

### 3.3.4 Internetmarkt

Mit dem Internet-Boom der letzten Jahre haben sich zahlreiche kommerzielle Anwendungen im Internet entwickelt. Wichtige Marktsegmente sind:

- E-Commerce (Online-Handel mit Produkten)
- E-Government (Online Abwicklung von Behördenwegen)
- Application Service Providing (Online-Software-Vermietung)
- Marktplätze (Auktionen, Matching-Funktion)
- Content (Portale, Communities, Content-Verkauf)
- Infrastruktur (Internet Service Provider, Telekommunikationsanbieter, Netzbetreiber)
- Mobile Business

In vielen dieser Märkte spielt OSS eine Rolle, jedoch in erster Linie als technische Lösung (z.B: Webserver oder Sicherheitslösung). Dies ist hier jedoch für die Marktanalyse nicht interessant, da OSS nur Mittel zum Zweck ist. Auch Internetseiten zum Thema OSS sind für die Marktanalyse nicht geeignet und werden somit in dieser Thesis nicht betrachtet.

Einzig im Marktplatzmodell ist OSS in einer spezifischen Rolle präsent. Im Marktplatzmodell findet ein zusammenführen von verschiedenen Interessensgruppen statt. Zusätzlich können sich Werbeanbieter beteiligen. Marktplätze nehmen die Rolle eines Vermittlers zwischen den verschiedenen Interessensgruppen wahr. Im Falle von OSS nennt man diese Marktplätze Mediatoren. Beispiele dafür sind Berlios ([www.berlios.de](http://www.berlios.de)) und Sourceforge ([www.sourceforge.net](http://www.sourceforge.net)).

Auch im kommerziellen / proprietären Softwareumfeld gibt es zahlreiche Marktplätze. Dort werden Teillösungen und Softwarekomponenten für alle Bereiche der Softwareerstellung angeboten.

Der Vorteil eines Marktplatzes liegt darin, dass die Kontaktkosten der Teilnehmer reduziert werden. Die Anwender sparen Zeit bei der Recherche nach einer bestimmten Software. Der Entwickler hat den Vorteil, ohne Streuverluste potentielle Kunden zu erreichen. Der Vorteil für Dienstleister

liegt wiederum darin, das Angebot möglichst vielen potentiellen OSS-Nutzern zu präsentieren.<sup>61</sup>

Die folgende Abbildung visualisiert das Zusammenspiel der Teilnehmer eines Marktplatzes:

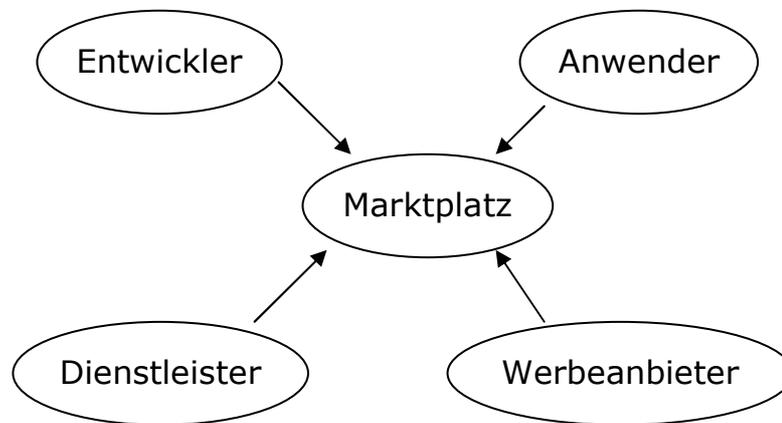


Abbildung 8: Zusammenspiel von Marktplatz-Teilnehmern

(aus: Leiteritz (2002), S. 27)

## 3.4 SWOT Analyse

### 3.4.1 Grundlegendes

Um eine Entscheidung für den Einsatz von OSS anhand einer sorgfältigen Analyse treffen zu können, sind insbesondere auch die Stärken, Schwächen, Potenziale und Bedrohungen zu berücksichtigen. Es wird in diesem Abschnitt hauptsächlich auf strategische und wirtschaftliche Gesichtspunkte eingegangen.

Angesichts des derzeitigen Hypes kann ein Gefühl der Begeisterung für OSS entstehen. Unternehmerische Entscheidungen sollten weder aus unbestimmbaren Gefühlen, noch aus der Nachahmung von Trends heraus getroffen werden.

---

<sup>61</sup> Vgl. Leiteritz (2002), S. 25 ff

Um eine Methodik für unternehmerische Entscheidungen bereit zu stellen, bietet sich die SWOT-Analyse<sup>62</sup> an. Damit wird untersucht, welche Argumente für und wider Freie- und Open-Source-Software in Unternehmen sprechen. Alle Aussagen, die im Folgenden getroffen werden, werden immer nur für einen Teil von OSS-Projekten Gültigkeit besitzen. Es lassen sich sicherlich für jedes Argument mehrere Gegenbeispiele finden. Daher muss im Unternehmen im Einzelfall abgewogen werden, welche Schritte in Richtung OSS aus unternehmerischer Sicht sinnvoll sind.<sup>63</sup>

### **3.4.2 Stärken**

Ohne unbestreitbare Stärken, würde OSS nicht auf so viel Interesse stoßen. Einer der wichtigsten Vorteile von Open Source Software besteht im „Peer Review“ Prinzip<sup>64</sup>. Darunter versteht man, dass der Source Code von anderen Experten ohne Zeitdruck durchleuchtet und verbessert wird. Je mehr Experten die Quellen kritisch überprüfen, desto weniger Fehler bleiben unbemerkt. Dies fördert die Stabilität, die ihrerseits wiederum ein kritischer Faktor für den Betrieb von IT-Systemen ist. Mangelnde Stabilität bewirkt Abstürze, Ausfallzeiten und hohen Wartungsaufwand. Daraus folgt, dass vor allem für Infrastrukturkomponenten wie Betriebssystemkerns, Treiber oder Netzwerkdienste das Open Source Modell sinnvoll ist. Aus dem Qualitätskriterium „Peer Review“ folgt, dass OSS im Allgemeinen eine höhere Sicherheit als proprietäre Software aufweist. Die Implementierungen werden von unabhängigen oder staatlichen Sicherheitseinrichtungen (z.B. Geheimdienste) überprüft und zertifiziert. Weiters folgt auch die Gesamtkonzeption bei freier Software meist einem strengeren Sicherheitsmodell. Dabei ist im Speziellen die IT-Sicherheit gemeint, denn gegen Fehler, die dem Faktor Mensch unterlaufen (z.B: Notieren von Passwörtern) sind IT-Systeme nicht in wirtschaftlich vertretbarem Maße absicherbar.

---

<sup>62</sup> SWOT = Strengths, Weaknesses, Opportunities und Threats

<sup>63</sup> Vgl. Wieland (2004), S. 107 ff

<sup>64</sup> Vgl. Raymond (1999)

Ein dritter Punkt ist die Offenheit des Quellcodes. Jeder kann ihn einsehen und Teile davon entnehmen – man muss das Rad nicht immer wieder neu erfinden. Dies ist mit proprietärer Software nicht möglich. Ebenso ist es schwieriger, unerwünschten Quellcode (z.B: Trojaner<sup>65</sup>, Virus) zu verbergen.

Darüber hinaus hat sich in der Praxis herausgestellt, dass entdeckte Fehler und Sicherheitsprobleme in etwa sechsmal schneller gelöst werden als bei proprietärer Software. Proprietäre Softwareanbieter versuchen zunächst den Fehler zu verharmlosen. OSS-Entwickler bieten hingegen rasche und punktuelle Hilfe und Problemlösungen.

Weiters bestehen Stärken dadurch, dass OSS auf bestimmte Anwendungsgebiete optimiert werden kann, wie Hochsicherheitsserver und es besteht keine Abhängigkeit von einem einzelnen Hersteller.

Die kostenlose Bereitstellung dieser Software hat im Wesentlichen dazu beigetragen, dass OSS bei den Unternehmen Interesse geweckt hat und laufend mehr Anwender findet. Dieser nicht zu unterschätzende Vorteil beinhaltet das effektive Einsparungspotential gegenüber der kommerziellen Software, was in den Kapiteln vier und fünf dieser Master-Thesis noch näher behandelt wird.<sup>66</sup>

---

<sup>65</sup> Trojaner sind bösartige Funktionen, die zur Spionage verwendet werden.

<sup>66</sup> Wieland (2004), S. 109 f

Die Umfrage der IT-Berater Cambridge Technologie Partners ergab folgende Vorteile von Open Source Software aus Sicht der zuständigen IT-Chefs in den 55 befragten Schweizer Unternehmen (Grad der Einverständnis):<sup>67</sup>

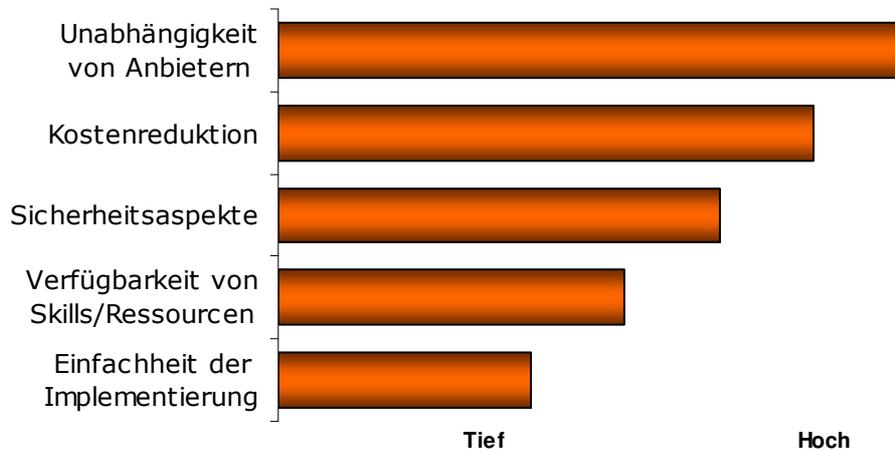


Abbildung 9: Argumente für den Einsatz von OSS (aus: Computerworld (2003), S. 3)

### 3.4.3 Schwächen

Sowie es Stärken gibt, existieren auch Schwächen bei OSS.

In erster Linie steht hinter einer Open Source Software in den seltensten Fällen eine Firma und man sollte nicht erwarten, dass man eine Anwendung in derselben Ausstattung bekommt wie eine kommerzielle Software. Den Entwicklern geht es eher um die Funktionalität als um die Ergonomie oder Dokumentation. Die letzten beiden Punkte kann sich der Anwender entweder selbst aneignen oder er kauft es als Service bei einer Distribution wie Suse oder Red Hat ein.

OSS hat im Vergleich zu proprietärer Software ein geringeres Angebot an Applikationen, insbesondere im Desktop-Bereich. In den letzten Jahren hat sich auf diesem Gebiet zwar viel bewegt, jedoch gibt es für einige kommerzielle Softwareprodukte keine OSS Pendant. OSS hat sehr gute Applikationen im Infrastrukturbereich (z.B: Webserver) jedoch Mängel bei Unternehmens- und Verwaltungsapplikationen (z.B: Workflowmanage-

<sup>67</sup> Vgl. Computerworld (2003), S. 3

ment) und Clientapplikationen (z.B: Datenaustausch mit kommerzieller Software)

Ein weiterer Punkt ist, dass die Softwareprodukte oftmals nicht den Reifegrad der kommerziellen Software erreicht haben. Den Versionsnummern sollte man hierbei weniger Vertrauen schenken, da sie nicht unbedingt aussagekräftig sind. Besser geeignet sind die sich etablierenden Techniken des Versionsmanagements. Wird ein Entwicklungsstand beispielsweise als „Stable Release“ bezeichnet, kann man im Allgemeinen von einer stabilen Version ausgehen. „Unstable“ oder „Snapshot“ ist dagegen eher mit Vorsicht zu begegnen und sind eher nicht für die Produktionsumgebung in Unternehmen geeignet. Um die am Besten geeignete Version für den Unternehmenseinsatz zu finden, empfiehlt es sich oft auf Dienstleister zurück zu greifen.

Ein dritter Punkt ist, dass es für eine Aufgabe mehrere Projekte und Lösungswege gibt, die sich stark in ihrem Reifegrad unterscheiden. Mögliche Kriterien für eine Auswahl können sein:

- Zugesicherte oder getestete Funktionalität
- Stabilität
- Qualität von Architektur und Design
- Lizenzmodell
- Unterstützung durch die Projekt Community
- Zahl der Mitarbeiter und Dauer des Projekts
- Engagement von Firmen.

Eine weitere Schwäche von OSS ist, dass viele Entwickler versuchen, sich mit den Entwicklern anderer OSS-Projekte zu koordinieren. Oftmals ist unbekannt, auf welchen OSS-Projekten die andere Software aufbaut. Es kann sein, dass Anwendungen in unterschiedlichen Versionen benötigt werden, um eine funktionierende Laufzeitumgebung für andere Programme zu schaffen. Im Extremfall kann dies soweit führen, dass sogar ein vollständiges Upgrade des Betriebssystems notwendig wird.

Eine weitere Schwäche sind die zahlreichen Updates bei OSS, da dies bei Unternehmen mit einem hohen Kostenaufwand und Unsicherheit verbunden sein kann.<sup>68</sup>

Die Umfrage der IT-Berater Cambridge Technologie Partners ergab folgende Nachteile in Prozent von Open Source Software aus Sicht der zuständigen IT-Chefs in den 55 befragten Schweizer Unternehmen (in Prozent der befragten Unternehmen):<sup>69</sup>

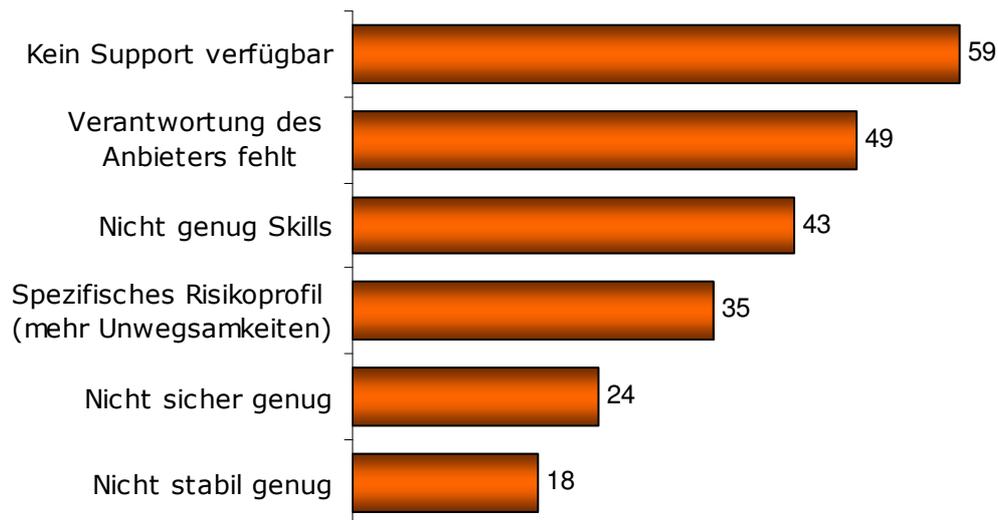


Abbildung 10: Gegenargumente für den Einsatz von OSS (aus: Computerworld (2003), S. 3)

### 3.4.4 Potenziale

Der auffälligste Vorteil von OSS ist, der Verzicht auf Lizenzgebühren. Weiters hat der Benutzer das Recht, sie zu kopieren, zu verbreiten, zu analysieren und zu verbessern. Heutzutage hat sich eingebürgert, dass OSS auf Datenträgern gegen Gebühr erhältlich ist, im Internet aber kostenlos verfügbar ist (abgesehen von Gebühren für den Internet Service Provider (ISP)). Daher kann man in der Praxis die Lizenzkosten für OSS vernachlässigen. Im Gegensatz dazu wird für kommerzielle Software eine Lizenzgebühr für jede Kopie verlangt.

<sup>68</sup> Vgl. Wieland (2004), S. 110 ff

<sup>69</sup> Vgl. Computerworld (2003), S. 3

Lizenzkosten machen nicht den Hauptteil der Gesamtkosten aus. Die Gesamtkosten werden oft auch als „Total Cost of Ownership“ (TCO)<sup>70</sup> bezeichnet. Zahlreiche Studien verglichen freie und proprietäre Software anhand der TCO Kriterien. Je nachdem, wer die Studie in Auftrag gegeben hat, für dessen Seite der Betrachtung kam immer ein Vorteil gegenüber der anderen Seite heraus. Somit variiert das Einsparungspotenzial von OSS im Gegensatz zu Microsoft Windows mit ca. 33% über einen 3-Jahres Zeitraum. Ebenso wurde auch ein Einsparungspotenzial von Windows gegenüber OSS von ca. 20% gefunden. Pauschale Modellrechnungen sind nur Anhaltspunkte, die im Allgemeinen wenig Aussagekraft haben und für jede Firma individuell analysiert werden müssen.

Neben den Kostenargumenten gibt es noch weitere strategische Aspekte, die für OSS sprechen. Sind in einer Folgeversion einer kommerziellen Software größere Änderungen in der Funktionalität vollzogen worden, so ist der Anwender gezwungen, die notwendigen Anpassungen in seiner Umgebung durchzuführen, sofern er dem Hersteller treu bleibt. Auch wenn die Unterstützung für ein Produkt wegfällt, ist der Anwender gezwungen, in Upgrade durchzuführen. Dies ist aktuell bei Microsoft Windows NT zu sehen, wo ein Plattformwechsel auf Windows XP vorgeschlagen wird. Hierbei könnte OSS als weiteres mögliches Umstiegszenario in Betracht gezogen werden.

Kritisch für Unternehmen kann es werden, wenn der Hersteller der verwendeten Software plötzlich in Konkurs geht oder von einer anderen Firma aufgekauft wird. In diesem Falle muss sich der Anwender um die Weiterexistenz der verwendeten Software fürchten, was beim Einsatz in kritischen Bereichen besonders ärgerlich ist. Diese Problematik fällt bei Open Source weg. Es gibt keinen direkten Hersteller und somit keine Abhängigkeiten und im Notfall kann man die Software selbst weiterpflegen.<sup>71</sup>

---

<sup>70</sup> Mehr darüber im Kapitel 4.2.1

<sup>71</sup> Vgl. Wieland (2004), S. 112 ff

Weitere Potentiale für OSS ergeben sich für Softwarehersteller, die vorhandene Programme als Open Source veröffentlichen und so der breiten Gemeinschaft zugänglich machen. Auf diese Potenziale wird aber nicht näher eingegangen.

### **3.4.5 Bedrohungen**

Die OSS-Gemeinschaft steht einigen Bedrohungen gegenüber.

Am Beginn war die Grundidee, freie Software zu entwickeln und von den Nutzern auch Unterstützung und Mitarbeit zu erhalten. In der letzten Zeit wurde dieses Prinzip durchbrochen, denn heutzutage gibt es viel mehr reine Nutzer als Entwickler. Jedoch ist die wachsende Verbreitung von OSS eher als ein Erfolg für die Leistungen der Entwickler zu sehen.

Dadurch kann sich die Motivation der Projektmitglieder ändern. Standen früher eher idealistische Beweggründe im Vordergrund, so spielen immer öfters auch materielle Gründe eine Rolle. Die Free Software Foundation schätzt, dass bereits mehr als ein Drittel der Entwicklungsarbeit in allen OSS-Projekten von Unternehmen bzw. deren Mitarbeitern geleistet wird.

Das OSS wieder vom Markt verschwindet ist keine realistische Gefahr. Es kommt jedoch oft vor, dass OSS-Projekte aus Mangel an Entwicklern eingestellt werden. Dies betrifft vor allem kleinere Projekte, die wenige Entwickler begeistern können und dadurch ihre Ziele nicht erreichen können.

Oftmals wird auch der „virale“ Charakter von OSS als Bedrohung empfunden. Es führt soweit, dass andere Software beeinflusst wird und Unternehmen keinerlei kommerziell vertriebene Software einsetzen. Dies ist jedoch völlig falsch und irreführend, wie Beispiele von IBM<sup>72</sup>, HP<sup>73</sup> oder Sun Microsystems<sup>74</sup> zeigen. Der Urheber hat das Recht zu entscheiden, unter welcher Lizenz seine Software vertrieben wird. Da es mehrere OSS-Lizenzmodelle gibt, kann der Entwickler selbst das beste Modell für seine Software auswählen.

---

<sup>72</sup> IBM (10.09.2004)

<sup>73</sup> HP (10.09.2004)

<sup>74</sup> Sun Microsystems (10.09.2004)

Eine weitere Schwierigkeit im Umgang mit OSS sind die gegensätzlichen Einstellungen, Motivationen und Erwartungen der Nutzer, der OSS-Entwickler und der firmeneigenen Entwickler. OSS-Projekte bestehen aus zufälligen Entwicklern, Dezentralisierung („Peer Review“) und bringen häufig neue Releases heraus. Praktiken von Unternehmen wie Aufgabenverteilung durch Projektleiter, fixe Terminpläne und Arbeitszeiten sind bei OSS-Projekten eher unüblich. Somit müssen sich Unternehmen, die OSS-Produkte einsetzen wollen, an die Kultur der Open Source Entwickler anpassen und ihre Spielregeln beherzigen. OSS-Entwickler akzeptieren im Allgemeinen nicht, dass jemand ihnen Entwicklungsprozesse aufzwingt.

Eine große Bedrohung für OSS ist die gegenwärtige Unsicherheit bei der Rechtslage. Einerseits gibt es in der EU immer wieder Diskussionen über Softwarepatente und andererseits gibt es schwellende Konflikte mit Urheberschaftsansprüchen. Die Einführung von Softwarepatenten könnte die OSS-Gemeinschaft in ernste Schwierigkeiten bringen, da das Patentgesetz Abgaben an den Patentinhaber inkludiert.

Die andere rechtliche Bedrohung ist die Urheberschaft für geistige Leistungen bei Codebestandteilen. Ein aktuelles Beispiel dafür ist die gerichtliche Auseinandersetzung der Santa Cruz Operation (SCO) mit IBM. Aus dieser Auseinandersetzung folgt, dass Linux eine Lizenzverletzung darstellt und somit Lizenzgebühren fällig werden. Es ist unwahrscheinlich, dass der Streit letztlich zugunsten von SCO ausgehen wird – zu viele Fakten sprechen dagegen. Er zeigt jedoch die Verletzlichkeit von Open Source Software.

Dass eine bestimmte OSS Rechte Dritter verletzt, die dann auf deren Einhaltung bestehen, ist nie ganz auszuschließen. Für den einzelnen Nutzer besteht dabei eher ein geringes Risiko.<sup>75</sup>

---

<sup>75</sup> Vgl. Wieland (2004), S. 115 ff

Eine ernstzunehmende Bedrohung ist das Forking, die Aufspaltung eines Projektes in mehrere Entwicklungsstränge, die meistens zueinander inkompatibel werden. Dadurch kann einerseits die Qualität der Software verbessert werden aber andererseits, was öfter vorkommt, kann das Projekt auseinander brechen und die Weiterentwicklung wird gestoppt.

Normalerweise stellen Mitarbeiter an OSS-Projekten ihre persönlichen Ressourcen nur solange zur Verfügung, bis ihre persönlichen Probleme mit dem Projekt gelöst sind. Wenn dieser Antrieb zur Weiterentwicklung wegfällt, müssen andere Anreize geschaffen werden. Daher sind OSS-Projekte auf Sponsoren und Förderungen der OSS Organisationen angewiesen, um ihr Überleben zu sichern, was wiederum zu Abhängigkeiten führt.<sup>76</sup>

---

<sup>76</sup> Vgl. Arpagaus/Seelhofer/Brüngger (2003), S. 88 ff

## 4. Wirtschaftlichkeitsbetrachtung

### 4.1 Ausgangsszenario

Die Durchführung einer Wirtschaftlichkeitsbetrachtung ist auf Grund der häufig multidimensionalen Wirtschaftlichkeitsmodelle für IT-Vorhaben nahezu unlösbar. Bei einer umfangreichen Analyse – wie es beim Vergleich von Microsoft- und OSS-Plattformen der Fall ist – gehört die Herstellung der Vergleichbarkeit der Untersuchungsobjekte sowie des richtigen Umfangs der Analyse zu den wesentlichen Aufgaben. Eine Betrachtung einzelner, isolierter Aspekte (z.B. nur der Serverbereich) lässt sich nicht auf das Gesamtergebnis übertragen und liefert im Allgemeinen ein falsches Ergebnis, was die unterschiedlichen Ergebnisse der Marktforschungsunternehmen erklärt.

In die Betrachtung mit einbezogen gehört auch die Berücksichtigung der Nutzerstruktur. Hierbei ist die Größe der Organisation (z.B: Unternehmen, Behörden, Vereine) und das unterschiedliche Ausgangsszenario der IT-Umgebung relevant bei der Wirtschaftlichkeitsbetrachtung einer Migration. Bei kleineren Organisationen kann die IT-Infrastruktur oftmals mit einfachen Mitteln aufgebaut und ohne intensive Ausbildung der Beteiligten betrieben werden. Hingegen erfordert der Betrieb der IT-Infrastruktur bei großen Organisationen gut ausgebildete Mitarbeiter, Service Level Agreements, organisatorische Regelungen für Problemfälle und eine andere Hardware-Ausstattung.

Unter Berücksichtigung der oben genannten Randbedingungen ist für die IT-Wirtschaftlichkeitsbetrachtung eine mehrdimensionale Betrachtung notwendig. Eine Erhöhung der Wirtschaftlichkeit kann auch durch personelle, organisatorische und rationalisierende Maßnahmen erreicht werden. Mit einer entsprechend ausgelegten IT-Strategie kann ebenfalls ein wesentlicher Beitrag zur Erhöhung der Wirtschaftlichkeit geliefert werden.<sup>77</sup>

---

<sup>77</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 293 f

Die Gesamtwirtschaftlichkeit der IT wird stark beeinflusst durch:

- *„Den Grad der funktionalen Abdeckung durch preiswerte Standardprodukte*
- *Qualität, Änderungsflexibilität und Entwicklungsfähigkeit der eingesetzten Standards, Technologien und Produkte*
- *Effizientes Einführungs- und Systemmanagement*
- *Bruchfreie Integration von Komponenten und Systemen in einer prozessorientierten Wertschöpfungskette*
- *Gute (interne oder externe) Service-Organisation sowie hochwertiges Know-how*
- *Wirtschaftliche Lebenszyklen der Produkte*
- *Kosten und Effizienz des Beschaffungsprozesses, sowie*
- *Wettbewerb bei Produkten und Dienstleistungen*<sup>78</sup>

Erst das optimale Zusammenspiel dieser Faktoren beeinflusst die Wirtschaftlichkeit insgesamt. Somit kann eine vereinfachte Betrachtung von Einzelkosten das Gesamtbild nicht korrekt widerspiegeln.

Neben der Ermittlung und dem Vergleich der Kosten ist auch die Beurteilung der möglichen Nutzenwerte ein wesentlicher Aspekt der Wirtschaftlichkeitsbetrachtung. Gerade hier spielen strategische Überlegungen eine wichtige Rolle, um einerseits die Ausgangssituation und andererseits die Perspektive ganzheitlich beurteilen zu können. Zum Beispiel kann eine teure Komponente Mehrkosten verursachen, aber in der strategischen Betrachtung aufgrund der Herstellerunabhängigkeit zu einem günstigeren Gesamtergebnis führen.

Sowohl die Methode als auch das Ergebnis kann aus diesen Gründen als Hilfsmittel bei der Ermittlung der eigenen Wirtschaftlichkeit und der Formulierung der eigenen IT-Strategie dienen.<sup>79</sup>

Im Folgenden wird nur die Wirtschaftlichkeit bei OSS- oder Windows-basierter Software betrachtet. Die anderen wichtigeren Betriebssysteme, wie UNIX oder MacOS sowie deren Applikationen werden nicht betrachtet.

---

<sup>78</sup> Bundesministerium des Innern - BMI(2003), S. 294

<sup>79</sup> Vgl. Bundesministerium des Innern - BMI(2003), S. 294

## **4.2 Methodische Grundlagen**

### **4.2.1 Allgemeines**

Ein Vergleich von funktional und qualitativ unterschiedlichen Dingen ist zwar grundsätzlich möglich, dies setzt aber eine Kosten-Nutzen-Analyse voraus, in der auch die zu erwartenden Produktivitätszuwächse den erwarteten Mehrkosten gegenüber gestellt werden. Die Produktivitätsanalyse der IT-Wertschöpfungskette würde den Umfang dieser Master-These sprengen und wird somit nicht durchgeführt. Sie würde beim Vergleich von OSS und CSS wahrscheinlich zu einem ausgewogenen Ergebnis führen, da es sich um reife Produkte mit langjähriger Entwicklungszeit handelt. Daher wird im Rahmen der Wirtschaftlichkeitsbetrachtung dieser These nur die direkte, vereinfachte Monetäre- und die Nutzwertanalyse näher betrachtet.<sup>80</sup>

### **4.2.2 Total Cost of Ownership**

Jeder erfahrende IT-Verantwortliche weiß, dass Lizenzkosten einer Software nur einen geringen Anteil an den Gesamtkosten ausmachen. Somit wurde vor einiger Zeit das Grundprinzip „Total Cost of Ownership“ (TCO) entwickelt. Bei diesem Grundprinzip werden die zurechenbaren Kosten in zwei große Gruppen aufgeteilt – direkte und indirekte Kosten.

Zu den direkten Kosten zählen alle budgetierbaren Kosten, d.h. sie sind ohne Umrechnung in monetären Einheiten messbar.

Die indirekten Kosten sind nicht budgetierbare Kosten. Dazu zählen Ausfallszeiten, unproduktive Benutzeraktivitäten, Selbsthilfe, formales und informelles Lernen, Datenverwaltung, Spielen, Internetsurfen und ähnliches.

---

<sup>80</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 294 f

TCO ist die Summe dieser Kosten für eine Investition in Software. Für die weitere Berechnung der Kosten nach TCO werden die Kosten für externe Kommunikation, Medien, Strom und Platzbedarf ausgeklammert, da sich diese bei den verschiedenen IT-Systemen kaum unterscheiden.

*„Es gibt einer ganzen Reihe von Studien, in denen die TCO von proprietärer und freier Software verglichen werden. Die meisten davon stellen GNU/Linux und Windows oder ein kommerzielles Unix gegenüber. So kommen beispielsweise Gillen et al. (2001) zu dem Schluss, dass GNU/Linux im Bereich Intranet/Internet nur 55% der TCO-Kosten pro Jahr verursacht, die ein vergleichbares System auf RISC/Unix-Basis mit sich bringen würde. Das australische Marktanalyse-Unternehmen Cybersource (2002) stellt bei einem ähnlichen Vergleich von GNU/Linux mit Windows eine Kostenersparnis von bis zu 34,3 % über einen Dreijahreszeitraum fest. Der Anwalt Brendan Scott (2002) hat mit eher theoretischen Überlegungen gezeigt, warum freie Software langfristig niedrigere Gesamtkosten haben muss. Und sogar der CEO von Sun Microsystems, Scott McNealy, spricht im Zusammenhang mit GNU/Linux immer wieder von bis zu 90% niedrigeren IT-Kosten (zitiert nach Nowak 2003).“<sup>81</sup>*

Welche Schlüsse kann man nun für die Praxis daraus ziehen? Es gibt ein beachtliches Einsparpotential durch OSS für große Unternehmen und ein kleineres Einsparpotential für kleine Unternehmen. Dies ist dadurch bedingt, da die Lizenzkosten für Hunderte von Arbeitsplätzen wegfallen und somit ein geringeres Einsparungspotential vorherrscht.

Jedoch sind pauschale Modellrechnungen nur Anhaltspunkte und besitzen eine begrenzte Aussagekraft. Für jede einzelne Firma muss separat untersucht werden, welche Kosten für freie oder proprietäre Software entstehen. Da bei dieser Methode ausschließlich die Kostenaspekte behandelt werden und eine Rentabilitätsanalyse fehlt, empfiehlt es sich, nach weiteren, besseren Methoden für eine Migrationsentscheidung zu suchen, was in der Folge durchgeführt wird.<sup>82</sup>

---

<sup>81</sup> Wieland (2004), S. 112 f

<sup>82</sup> Vgl. Wieland (2004), S. 112 f, Bundesministerium des Innern – BMI (2003), S. 297

### 4.2.3 IT-Wirtschaftlichkeitsbetrachtung

Die deutsche Koordinierungs- und Beratungsstelle der Bundesregierung für Informationstechnik in der Bundesverwaltung (KBSt) hat bereits 1992 eine Handlungsanweisung für die Wirtschaftlichkeitsbetrachtung der IT in der Verwaltung (IT-WiBe) erarbeitet. Diese Anweisung wurde immer wieder aktualisiert und liegt nun in der dritten Version vor.

Im Wesentlichen beinhaltet die IT-WiBe drei Schritte:

- Einflussgrößen festlegen (Kriterien selektieren)
- Daten erheben/bewerten
- Kennzahlen ermitteln

Anders als TCO ist IT-WiBe ein Verfahren, das nicht nur den Kostengesichtspunkt betrachtet, sondern auch mögliche Ersparnisse in die Berechnung miteinbezieht, wie folgende Abbildung zeigt.<sup>83</sup>

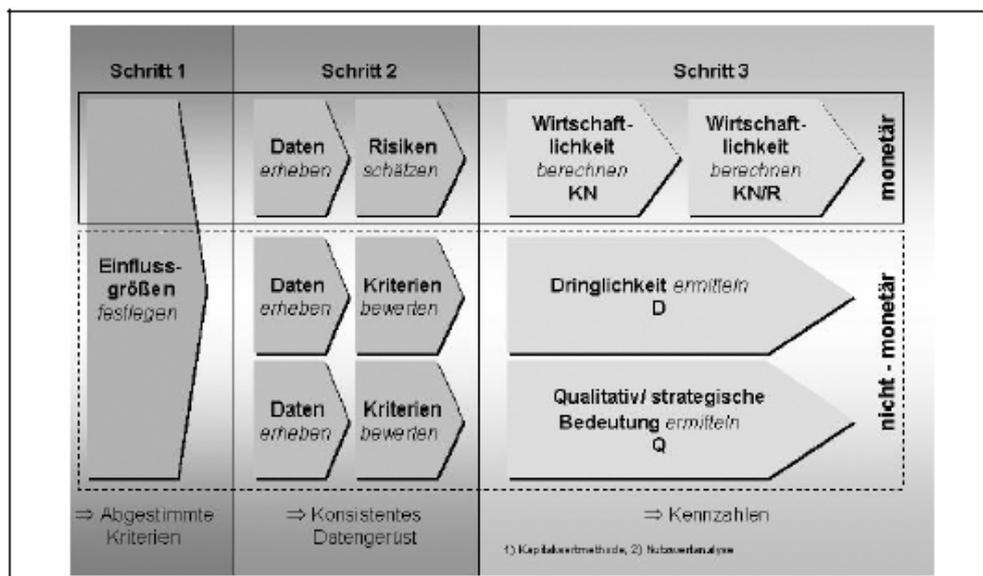


Abbildung 11: IT-WiBe-Methodik (aus: Bundesministerium des Innern - BMI (2003), S. 296)

### 4.2.4 Monetäre Analyse

Als Methodik für die monetäre Analyse wird die Kapitalwertmethode angewandt.

Als dynamisches Verfahren beurteilt sie die Investitionsprojekte nach ihrem Kapitalwert. Dadurch wird eine wirklichkeitsnahe Erfassung der mit

der Investition zusammenhängenden Finanzströme (Einnahmen und Ausgaben, haushaltswirksam und nicht haushaltswirksam), fokussiert auf einen gemeinsamen Bezugszeitpunkt erreicht. Einnahmen und Ausgaben, die mit den Vorhaben zusammenhängen, können für fünf Jahre im Voraus geplant werden. Für die zukünftigen Werte wird der aktuelle Zeitwert durch Abzinsung mit einem Zinssatz von fünf Prozent<sup>84</sup> ermittelt.<sup>85</sup>

#### **4.2.5 Migration-Kosten-Matrix**

Obwohl die IT-WiBe für eine projektbezogene Wirtschaftlichkeitsbetrachtung gut geeignet ist, ist sie für ein Gesamtmodell häufig zu komplex und nicht durchführbar. Als eine vereinfachte Methode gilt die Migrationskostenmatrix. Diese Methodik verzichtet auf die nach selektiven Kriterien ermittelten Kosten und Ersparnisse sondern aggregiert diese in drei Kategorien – Hardware, Software und Personal.

In diesen Kategorien werden für den Untersuchungszeitraum (z.B: 5 Jahre) die Beschaffungs- und Folgekosten<sup>86</sup> als auch die möglichen Ersparnisse erfasst. Eine Gesamtübersicht liefert die Daten über die untersuchten Jahre und Kategorien sowie eine Rentabilitätsbetrachtung auf einen Bezugszeitpunkt.

Diese Methodik liefert Praktikern ein Instrument, mit dem einfach und schnell ein Projektkostenvolumen inkl. Folgekosten und Ersparnissen ermittelt werden kann.<sup>87</sup>

Eine Abbildung der Migrations-Kosten-Matrix ist im Abschnitt 4.3.2 – Kostenkategorien – enthalten.

---

<sup>83</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 295 f

<sup>84</sup> Kann jederzeit an unternehmenseigene Zinssätze angepasst werden

<sup>85</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 295

<sup>86</sup> Die Kostenbereiche Beschaffung und Folgekosten sind ebenso wie die Ersparnisse mit folgenden Unterbereichen versehen: Server-Infrastruktur, Datenbank-Anwendungen, Messaging/Groupware, Web-Anwendungen, Office/ Desktop und Sonstiges.

<sup>87</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 296 f

## 4.2.6 Vergleichbarkeit

Um eine Vergleichbarkeit der unterschiedlichen Auswertungen zu gewährleisten, wird die Wirtschaftlichkeitsbetrachtung in zwei Szenarien durchgeführt:

1. Teilmigration: Migration einzelner oder mehrerer Migrationsobjekte bei klar abgegrenzten Produkten oder Produktgruppen
2. Gesamtmigration: Migration der kompletten IT-Umgebung – Server, Client, Infrastruktur, Applikationen

Die Migration kann nach folgenden zwei Möglichkeiten durchgeführt werden:

1. Ablösende/Vollständige Migration: Migration in eine komplett neue, OSS-basierte Software-Umgebung
2. Fortführende Migration: Migration im Rahmen der eingesetzten Produkte auf neue Versionen derselben.

Für die Teilmigration wird ein reduzierter Kriterien-Katalog eingesetzt, der speziell auf die Fälle der Teilmigration zugeschnitten ist. Die Gesamtmigration wird mit den allgemeinen Bewertungskriterien der IT-WiBe durchgeführt. Wenn Fachanwendungen umzustellen sind, wird auch die Entwicklungsarbeit für die speziellen Programme analysiert.

Werden nur einzelne Produkte oder Produktgruppen (z.B: MS Office) migriert, wird wiederum der spezifische Kriterienkatalog verwendet. Bei umfangreichen Produktgruppen wird der komplette IT-WiBe Kriterienkatalog verwendet.

Eine vergleichende Wirtschaftlichkeitsanalyse macht nur bei technisch und funktional vergleichbaren Alternativen Sinn. Somit ergeben sich folgende Einsatzbereiche der OSS- und Microsoft-Technologie:

- Infrastruktur-Dienste
  - Datei-Server
  - Druck-Server
  - Anmeldungs-Server

- Netzwerke
  - Messaging- und Groupware-Systeme
  - Office-Pakete
  - Datenbank- und Web-Anwendungsserver

Die IT-Sicherheit wird nicht verglichen, da es aus heutiger Sicht wegen der eindeutig höheren Gefährdung von Windows-Systemen nicht zielführend ist. Auch mit einem hohen Aufwand kann eine vergleichbare Absicherung der Systeme nicht erreicht werden, daher wird auf einen Vergleich im Sicherheitsbereich verzichtet.<sup>88</sup>

#### **4.2.7 Neueinführung vs. Migration von Systemen**

Bei der Kostenbetrachtung im Hinblick auf die Einführung neuer Technologien muss grundsätzlich zwischen einer Neueinführung und der Migration unterschieden werden. Die „Daumenregel“ besagt, dass grundsätzlich eine Neueinführung einfacher und billiger zu bewerkstelligen ist als eine Migration, bei der auf eine historisch gewachsene IT-Infrastruktur Rücksicht genommen werden muss, ohne dass Daten verloren gehen und der laufende Betrieb beeinträchtigt wird.

Da ein Migrationsverfahren grundsätzlich von seiner Ausgangssituation abhängt, ist eine allgemeingültige und allumfassende Aussage zu den Kosten kaum möglich. Insbesondere die Existenz von selbst entwickelten Anwendungen, die ebenfalls abgelöst werden müssen, Überführung von Altdaten, spezielle Nutzer- und Zugriffsrechte oder andere Besonderheiten führen zu einem erheblichen Projektaufwand, der unternehmensspezifisch untersucht werden muss.<sup>89</sup>

---

<sup>88</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 296 ff

<sup>89</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 299

## 4.2.8 Vollkostenansatz

Der Vollkostenansatz konzentriert sich im Wesentlichen auf folgende Alternativen:

- Open Source Software (OSS)
- Commercial Linux Software (COLS)
- Proprietäre Software (Microsoft Software (MS))

Die Ergebnisse geben einen grundsätzlichen Ausblick auf die langfristige Kostenentwicklung aus heutiger Sicht. Um die Entscheidungsgrundlage über einen potentiellen Wechsel und eine Migration zu vervollständigen, müssen die Kosten der Migration ermittelt werden. Mit Hilfe des Dienstleistungssektors kann eine solche Schätzung in Form eines Migrationsangebotes eingeholt werden und den ermittelten Potentialen gegenüber gestellt werden. Die dabei verwendete Methode berücksichtigt die inhomogene Struktur und die unterschiedliche Größe der Unternehmen. Die einzelnen Schritte zur Bestimmung der Vollkosten der einzelnen Alternativen sind:

- Definition der zu untersuchenden Einsatzfelder
- Ermittlung von Kostenfaktoren in den untersuchten Einsatzfeldern
- Ermittlung von Werten der Kostenfaktoren für:
  - Kleine Unternehmen
  - Mittlere Unternehmen
  - Große Unternehmen
- Ermittlung der Kosten für Rationalisierungsmittel (System-Management-Werkzeuge)
- Ermittlung der Kostenstruktur der untersuchten Alternativen
- Ermittlung der qualitativen und technischen Vergleichbarkeit
- Durchführung der Szenario-Analyse beim Wechsel:
  - OSS - Open Source Software
  - COLS - Commercial Linux Software

Das direkte Potenzial von OSS ergibt sich vor allem aus der Einsparung der Lizenzkosten. Die Analyse von langfristigen Potenzialen (Ermittlung

der Software-Lizenzkosten im Vergleich zu den Vollkosten der untersuchten Infrastrukturen) führt zum Ergebnis der Wirtschaftlichkeitsanalyse.<sup>90</sup>

#### **4.2.9 Nutzwertanalyse**

Gilt es nicht monetär erfassbare Auswirkungen in die Entscheidungsfindung einzubeziehen, so wird die Nutzwertanalyse angewandt. Sie bewertet einzeln und unabhängig voneinander gewichtete Zielkriterien, um sie anschließend zu einer Gesamtbewertung zusammenzufassen. Damit werden die so genannten „weichen“ Faktoren über Bewertungsskalen quantifiziert.

Es wird empfohlen folgendermaßen vorzugehen:

1. Die Ergebnisse der monetären Wirtschaftlichkeitsanalyse haben Vorrang
2. Die Ergebnisse der Nutzwertanalyse führen zu Dringlichkeits- und Strategiekennzahlen

Die Wirtschaftlichkeitsanalyse liefert die Kosten und Ersparnisse der Vorhaben, die in einer Rentabilitätskennzahl dargestellt wird und als positiver Kapitalwert ausgedrückt wird. Der zweite Schritt – die Nutzwertanalyse – dient in erster Linie dazu, dass eine monetäre Wirtschaftlichkeitsanalyse grundsätzlich nicht ausreicht oder keine eindeutige Rentabilitätsaussage liefert. Daher kann ein Vorhaben durch den zweiten Schritt eine hohe Priorität zur Durchführung erhalten, unabhängig von monetären Kriterien.<sup>91</sup>

---

<sup>90</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 299 f

<sup>91</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 295

## **4.3 Monetäre (operative) Dimension**

### **4.3.1 Einsatzbereiche**

Für ein aussagekräftiges Ergebnis wird die Gesamt-Analyse in mehreren Einsatzbereichen durchgeführt. Die untersuchten Kosten werden dabei in den folgenden Einsatzfeldern ermittelt:

- Server-Infrastruktur
  - Dateidienste
  - Druckdienste
  - Anmeldedienste
  - Netzwerkdienste
- Desktop-Infrastruktur
  - Office
  - Web
- Messaging/Groupware
- Datenbank- und Web-Anwendungen

Diese Aufzählung hat nicht die Vollständigkeit zum Ziel, sondern sollte ein gemeinsamer Nenner für die meisten Infrastruktur-Bereiche in Unternehmen sein.<sup>92</sup>

### **4.3.2 Kostenkategorien**

Für die Vergleichbarkeit der Ergebnisse der Kostenbetrachtung ist ein durchgängiges und eindeutiges Kostenmodell notwendig. Daher werden für das gemeinsame Kostenmodell folgende drei Kostenkategorien gebildet:

- Hardware
  - Vergleich der Hardware-Anforderungen
- Software
  - SW-Lizenzkosten
  - SW-Wartungskosten

---

<sup>92</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 300 f

- Zusatzkosten für Directory-Systeme
- Zusatzkosten für System Management und Sicherheit
- Personal
  - Administration
  - Support
  - Software-Pflege
  - Schulung

Behörde					
- Bezeichnung		Gesamt	Hardware	Software	Personal
- Anzahl User					
- Anzahl Standorte					
- Anzahl Server					
- Startjahr <sup>(1)</sup> 2003					
1. Jahr - 2003	Mengen- einheit <sup>(2)</sup>	Gesamt Aufwand <sup>(4)</sup> gesamt	Hardware Aufwand <sup>(4)</sup> gesamt	Software Aufwand <sup>(4)</sup> gesamt	Personal Aufwand <sup>(4)</sup> gesamt
Saldo		0	0	0	0
Kosten (Besch.+Folge)		0	0	0	0
Beschaffungskosten		0	0	0	0
Server-Infrastruktur		0	0	0	0
- Server		0	0	0	0
- Arbeitsplatzrechner		0	0	0	0
- Netzwerk		0	0	0	0
Datenbank-Anwendungen		0	0	0	0
Messaging/Groupware		0	0	0	0
Web-Anwendungen		0	0	0	0
Office / Desktop		0	0	0	0
sonstiges		0	0	0	0
Folgekosten		0	0	0	0
Einsparungen		0	0	0	0

Abbildung 12: Migrations-Kosten-Matrix mit Kostenkategorien und Einsatzfeldern (aus: Bundesministerium des Innern - BMI (2003), S. 302)

Hinweis zur Ausfallszeit: Nach vorliegenden Erfahrungen der Rechenzentren kann für Linux-Systeme eine höhere Verfügbarkeit als für Windows-Systeme angegeben werden. Somit hat ein Linux-System eine höhere produktive Auslastung.<sup>93</sup>

### 4.3.3 Eigenschaften der Unternehmensgrößen

In den folgenden Abschnitten werden die Eigenschaften unterschiedlicher Unternehmensgrößen stichpunktartig beschrieben. Aufgrund der Unterschiede bei der IT-Ausstattung und Organisation kann dies als grobe Ori-

<sup>93</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 301 f

entierungshilfe hinsichtlich der relevanten Untersuchungskriterien verwendet werden.<sup>94</sup>

#### 4.3.3.1 Kleines Unternehmen

- Nutzer: bis zu 100
- Hardware: i.d.R. kleine und preiswerte Intel-Plattform
- Backup und Recovery: preiswerte Backup-Mechanismen, Einsatz von Bandlaufwerken oder RAID-Systemen
- Personal: i.d.R. ein Administrator mit Universalprofil, ein Stellvertreter
- IT-Organisation: Einzelperson, ggf. Gruppe, geringer Spezialisierungsgrad
- Sicherheit und Verfügbarkeit: i.d.R. niedrige bis mittlere Anforderungen
- Systemmanagement: Einzelne Tools (MS) oder Skripte (Linux)

#### 4.3.3.2 Mittleres Unternehmen

- Nutzer: 100 bis 500
- Hardware: kleine und große Server-Systeme, Intel- und RISC Plattformen, sowohl dezentrale als auch zentrale Architekturen möglich
- Backup- und Recovery: dedizierte Backup- und Archiv-Server vorhanden, Einsatz der RAID-Technologie ist Regel
- Personal: Mehrere Administratoren im 8-Std. Betrieb, Spezialisierung, Bereitschaftsdienst
- IT-Organisation: IT-Abteilung
- Sicherheit und Verfügbarkeit : i.d.R. mittlere bis hohe Anforderungen
- Systemmanagement: SW-Verteilungstools, Thin Clients, Netzwerk- und Systemüberwachung

---

<sup>94</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 302 f, mit eigener Annahme

#### 4.3.3.3 Großes Unternehmen

- Nutzer: ab 500
- Hardware: preiswerte Intel-Cluster, große Server-Lösungen, verteilte Architekturen, zentrale Mainframe-Systeme
- Backup und Recovery: Zentrale Backup-/Disaster-Recovery-Server mit Roboter- oder Jukebox-Hardware
- Personal: Administratoren im 8-Std. Betrieb, Spezialisierung, Bereitschafts- und Notdienst
- IT-Organisation: Rechenzentrum, ggf. lokale Administrationsgruppen
- Sicherheit und Verfügbarkeit: Hohe bis sehr hohe Anforderungen, Einsatz von umfangreichen Storage Area Network (SAN) Lösungen
- Systemmanagement: SW-Verteilungstools, Thin Clients, Netzwerk- und Systemüberwachung

## 4.4 Strategische Dimension

### 4.4.1 Makroökonomische Bedeutung

Zusätzlich zur monetären Analyse der einzelnen Alternativen ist es notwendig, eine strategische Betrachtung durchzuführen. In der IT-WiBe Terminologie wird dies auch Dimension genannt.

Die Notwendigkeit ergibt sich aufgrund des strategischen Faktors „Herstellerabhängigkeit“ und hat sowohl makro- als auch mikroökonomische Bedeutung.

Bei der makroökonomischen Bedeutung spielen die wettbewerbsbedingten Aspekte eine besondere Rolle.

Die Vorteile eines funktionierenden Wettbewerbs sind:

- Bessere Produktqualität
- Niedrigere Produktpreise
- Höhere Innovationsrate

Alle Softwareproduzenten beanspruchen sowohl die höhere Produktivität als auch die technologische Innovation für ihre Produkte. In der Realität kann eine allgemeine Aussage jedoch nicht getroffen werden.

Die makroökonomische Betrachtung ist grundsätzlich bedeutend, jedoch ist sie nicht direkt durch die vorliegende Thematik beeinflussbar sondern eher durch unternehmenspolitische Entscheidungen. Daher wird sie auch in weiterer Folge nicht weiter vertieft.<sup>95</sup>

#### **4.4.2 Mikroökonomische Bedeutung**

Bei der mikroökonomischen Bedeutung spielt die Betrachtung der eigenen Abhängigkeit von Lieferanten, Produkten und Dienstleistungen eine Rolle. Sowohl bei Quasi-Monopolen als auch im funktionierten Wettbewerb kann eine starke Abhängigkeit von Lieferanten unter Umständen zu wirtschaftlichen Nachteilen führen.

Dies kann sich einerseits in höheren Produktpreisen und andererseits in kürzeren Lebenszyklen der Produkte niederschlagen. Im Extremfall kann die Abhängigkeit zu einer Situation führen, bei der keine preislich akzeptable Handlungsalternative mehr vorhanden ist. Eine auf strategischem Gleichgewicht aufgebaute Situation führt im Gegensatz dazu zu einer besseren Verhandlungsposition, da im Problemfall auf Alternativen zurückgegriffen werden kann.<sup>96</sup>

#### **4.5 Gesamtergebnis**

Die meisten themenspezifischen Untersuchungen verwenden den Vollkostenansatz für eine Analyse der Alternativen. Im Allgemeinen erhalten sie als Ergebnis, dass die Software-Lizenzkosten nicht den Hauptanteil der Kosten ausmachen, sondern die Personalkosten während der Einführung und beim Betrieb.

---

<sup>95</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 304

<sup>96</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 304

Aufgrund unterschiedlicher Annahmen, wie bei Administrierbarkeit und Sicherheit kommen bei den Untersuchungen unterschiedliche Ergebnisse bei den wirtschaftlichen Vorteilen der Alternativen heraus.

Bei Microsoft-vorteilhaften Untersuchungen werden die niedrigeren TCO zum größten Teil durch niedrigere Personalkosten und weniger aufwändige IT-Sicherheit erzielt. Die Personalkosten entstehen dadurch, dass Microsoft-basierte Systeme einen niedrigeren notwendigen Ausbildungsstandard des Personals benötigen und somit niedrigere Gehälter bezahlt werden. Bei der Sicherheit wird angenommen, dass sie insgesamt weniger aufwändig sei, was wiederum sehr umstritten ist.

Bei den Microsoft-kritischen Untersuchungen wird insbesondere die Administrierbarkeit als Pluspunkt vermerkt. Obwohl auch hier festgestellt wird, dass ein Unterschied bei den Gehältern besteht, wird dieser Unterschied durch die bessere Administrierbarkeit mehr als wettgemacht.<sup>97</sup>

Die Untersuchung im Rahmen dieser Thesis verweist im Wesentlichen auf folgende ausschlaggebende Faktoren für die Überlegungen aus Sicht der (für eine OSS-Einführung) notwendigen TCO-Betrachtung:

1. Anteil der Softwarelizenzkosten an den Gesamtkosten
2. Spezialisierungsgrad der in Frage kommenden IT-Systeme und Infrastrukturen
3. Automatisierungsgrad der in Frage kommenden IT-Systeme und Infrastrukturen

Der Anteil der Software-Lizenzkosten an den Gesamtkosten der IT-Systeme liegt zwischen 20% und 50%. Daher bewegt sich in diesem Bereich das unmittelbare Potential für OSS, gemessen in der rein monetären Dimension und in der Voraussetzung der Vergleichbarkeit der Arbeitsergebnisse.

---

<sup>97</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 304 ff

Für die weitere Vorgehensweise bei der Betrachtung der Alternativen sind zwei weitere Faktoren wichtig, einerseits der Index der direkt beeinflussbaren Kosten und andererseits die Analyse der haushaltswirksamen Kosten.

*„Zu den unmittelbar direkt beeinflussbaren Kostenarten zählen:*

- *Beschaffungskosten Software: im Wesentlichen durch Wechsel auf preiswertere Produkte oder Verhandlung günstigerer Beschaffungspreise*
- *Wartungskosten Software: im Wesentlichen durch Konsolidierung (Reduktion der Produktvielfalt) der Softwareprodukte oder Verzicht auf Update-Zyklen*
- *Beschaffungskosten Hardware: durch Wechsel auf preiswertere HW-Plattformen*
- *Wartungskosten Hardware: durch Konsolidierung (Reduktion der Produktvielfalt) der Hardware oder Verlängerung der Lebenszyklen<sup>98</sup>*

In der Betrachtung der direkt beeinflussbaren IT-Kosten wird deutlich, dass die Lizenzkosten (SW-Beschaffung, -Wartung, -Updatedurchführung) der größte Teil sind und somit den größten Handlungsspielraum aufweisen.

Der größte Block der IT-Ausgaben, die Personalkosten, zählt in der Regel nicht zu den direkt beeinflussbaren Größen. Dies resultiert daraus, da die Einführung und der Betrieb von IT-Infrastruktur und IT-Systemen mit einer Grundlast verbunden sind, die weniger durch die einzelnen Lizenzmodelle bestimmt wird. Vielmehr resultieren die Personalkosten aus der Betreuungsintensität, Verfügbarkeit und Sicherheit der betriebenen Plattformen. Die Reduktion von existierenden Personalkapazitäten oder deren Auslagerung und Konsolidierung stellen in der Regel keinen kurzfristig möglichen Handlungsspielraum dar.<sup>99</sup>

---

<sup>98</sup> Bundesministerium des Innern – BMI (2003), S. 305 f

<sup>99</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 304 ff

# 5. Migration

## 5.1 Grundlegendes

Unter Migration ist die Umstellung eines Systems auf ein Anderes zu verstehen. Dies kann sich sowohl auf ein gesamtes IT-System als auch auf Einzelbereiche beziehen. Einzelbereiche können Server, Arbeitsplatzrechner oder einzelne Softwareprodukte sein.

Eine wichtige Voraussetzung für das Gelingen einer Migration ist die fundierte Kenntnis über den Aufbau des alten und neuen Systems sowie die Anforderungen an dieses.<sup>100</sup>

Um eine Migration durchzuführen, bietet sich ein sechs Phasen-Modell an, welches im Folgenden stichwortartig skizziert wird:

1. *„Workshop: (Kick Off, betroffene Fachabteilungen und IT- Disziplinen beteiligen, Identifizieren aller relevanten Themen, Prioritäten setzen, Entscheidungsbedarf identifizieren, Vorgehensweise und Projektplan festlegen, detaillierte Aufwandschätzung, Teilprojekte festlegen und Arbeitsgruppen zuweisen)*
2. *Ist-Aufnahme: (Anwendungslandschaft, Kommunikationsbeziehungen, Netzwerkinfrastruktur, Zentrale Dienste, Betriebsverfahren, Zukünftige Anforderungen)*
3. *Grobkonzept: (Pflichtenheft erstellen, Projektplan verfeinern und Arbeitspakete definieren, Technische Machbarkeit, Aufbau einer Integrations- und Testumgebung, Abbildung der übrigen Produktionsumgebung, Anwendungsintegration, Hardware-Auswahl und Evaluierung)*
4. *Feinkonzept: (Detaillierte Festlegung des Funktionsumfangs, Integration in die übrige IT-Umgebung, Entwicklung der Installationsverfahren und Softwareverteilung, Integration in den Betrieb, Rollout-Planung, Pilotplanung, Ausbildung des DV-Personals)*
5. *Pilot: (Feature Stop, Repräsentative Benutzergruppe versorgen, Lasttests, Einbindung des UHD (User Help Desks), erstmalige Sizing-Kontrolle, Rückkopplung in Feinkonzept)*
6. *Rollout: (Inbetriebnahme der Installationsverfahren, Serversysteme vervielfältigen, Benutzerinformation und -Schulung, Begleitung durch Projektteam, Übergabe in den Regelbetrieb)“<sup>101</sup>*

---

<sup>100</sup> Vgl. BMWi/Innominate AG (2003), S. 30

<sup>101</sup> Bundesministerium des Innern – BMI (2003), S. 314

## 5.2 Auswirkungen

Die Umstellung auf ein neues System kann unterschiedliche Auswirkungen haben. Die Wichtigsten sind unten angeführt:<sup>102</sup>

- **Ausfallzeiten:** Bei der Umstellung kann das Gesamtsystem vorübergehend stillgelegt sein. Größere Ausfallzeiten können vermieden werden, wenn zuvor ein Testsystem aufgebaut wird, um die Software intensiv zu testen. So kann die Software im Allgemeinen störungsfrei eingeführt werden.
- **Datenkonvertierung:** Wenn die Desktopcomputer z.B. auf neue Office-Pakete umgestellt werden, müssen die Daten konvertiert werden. Es ist sinnvoll, die Daten in ein offenes Datenformat zu konvertieren, um mit möglichst vielen Produkten kompatibel zu sein.
- **Schulung:** Bei einer Systemumstellung ist im Allgemeinen auch mit einem Schulungsbedarf der Anwender zu rechnen. Bei einer Server-Umstellung gilt es die Systemadministratoren zu schulen, bei einer Client-Umstellung auch die Endanwender.

## 5.3 Migrationswege

### 5.3.1 Schnelle Migration

Die schnelle Migration ist nicht durch ihre Schnelligkeit geprägt, sondern durch den überschaubaren und festgelegten Zeitraum, so dass die Migration in einem Schritt durchgeführt werden kann. Eine schnelle Migration hat einen definierten Beginn und ein definiertes Ende. Da am Ende die Produktionsumgebung vollständig Linux-basiert ist, ergeben sich für die Durchführung hohe Ansprüche an folgende Bereiche:

- die Organisation des Projektes
- die Organisation des betroffenen Unternehmens
- die Technik
- die Finanzen

---

<sup>102</sup> Vgl. BMWi/Innominate AG (2003), S. 31

- die Administratoren
- die Benutzer

Von den oben genannten Punkten sind besonders die letzten Beiden wichtig. Die Anforderungen an die Administratoren und Benutzer dürfen nicht unterschätzt werden, insbesondere wenn diese über wenig Know-how bezüglich der neuen IT-Landschaft verfügen. Ein Vorteil der schnellen Migration ist, dass sich Administratoren nur kurze Zeit mit zwei unterschiedlichen IT-Systemen auseinandersetzen müssen.

Auch die Verfügbarkeit der benötigten Finanzmittel im relativ kurzen Zeitraum ist wichtig. Damit wird die Machbarkeit der schnellen Migration entschieden.

Die Anforderungen an die Organisation beziehen sich vor allem auf die Qualifizierung der Mitarbeiter, die ja weiterhin ihren täglichen Arbeiten nachgehen müssen. Es müssen die Störungen des Betriebsablaufes minimiert werden und der laufende IT-Betrieb muss aufrecht gehalten werden. Dies kann durch geeignete Umstellungskonzepte, welche für jedes Unternehmen spezifisch anzufertigen sind, gelöst werden.

Aufgrund der hohen Anforderungen an die schnelle Migration stellt sich die Frage, warum ist sie sinnvoll und zu empfehlen?

Folgende Gründe kann man anführen:

- Es besteht der Zwang zu einer Migration, das heißt, dass zum Beispiel der Support für die alten Systeme ausläuft
- Die Administratoren und Benutzer werden zwar intensiv, dafür aber nur einmal mit Neuerungen konfrontiert und nicht jährlich fortlaufend
- Die Administratoren müssen sich nicht über längere Zeiträume mit der Komplexität heterogener Welten auseinandersetzen

Welche Voraussetzungen sind für eine schnelle Migration sinnvoll?

In erster Linie, wenn eine relativ einheitliche Systemlandschaft vorliegt und nur wenige Anwendungen und Dienste umgestellt werden müssen. Dabei ist die Organisationsgröße nicht bedeutend. Eine weitere gute Voraussetzung ist, wenn die Administratoren über das notwendige Know-how bereits verfügen (z.B. sie beschäftigen sich in ihrer Freizeit mit OSS-Systemen). Wenn auch die Mitarbeiter die nötige Offenheit für eine Neuerung und Umstellung auf OSS mitbringen, steht einer schnellen Migration kaum mehr was im Wege.<sup>103</sup>

### **5.3.2 Sanfte Migration**

Im Gegensatz zur schnellen Migration ist bei der sanften Migration nur das Ziel feststehend. Der Weg zu diesem Ziel ist nur ein grob definierter Zeitplan und die Migration wird komponentenweise durchgeführt.

Die Gründe für die sanfte Migration können in den hohen Anforderungen für eine schnelle Migration liegen:

- In Unternehmen mit kleinen Budgets können die notwendigen Kosten an die Ertragssituation angepasst werden.
- Ein fehlendes Know-how kann schrittweise aufgebaut werden und somit Kosten eingespart werden. Da die Schulungen immer komponentenbasiert sind, ist beim nächsten Schritt ein höheres Know-how verfügbar.
- Bestehende Widerstände können langsam abgebaut werden und die Mitarbeiter können sich an die neue Umgebung gewöhnen
- Komplexe IT-Strukturen können nach und nach in verbesserte einfache Strukturen übergeführt werden.

---

<sup>103</sup> Bundesministerium des Innern – BMI (2003), S. 383 ff

Nachfolgende Abbildung zeigt einen möglichen Weg einer sanften Migration.

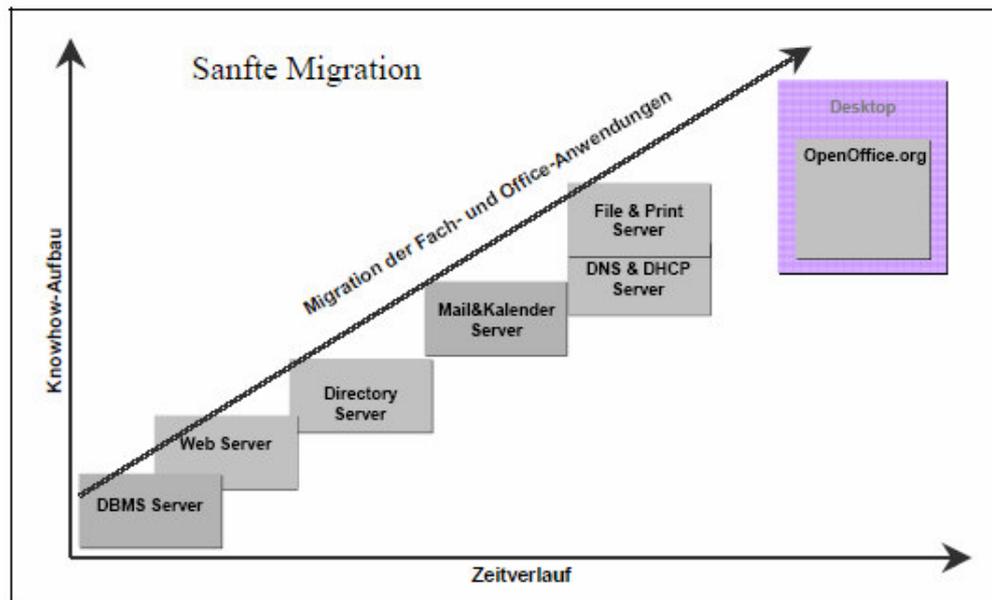


Abbildung 13: Sanfte Migration (aus: Bundesministerium des Innern - BMI (2003), S. 386)

Am Beginn sollte eine einfach herauszulösende Komponente gewählt werden. In diesem Beispiel ist es der DBMS Server. Es geht hierbei um das Aufsetzen eines parallelen Servers und nicht um die Migration der Datenbankanwendungen. Für die Migration des Web Servers werden in der Regel Grundkenntnisse des DBMS Servers benötigt. Der Directory Server kann als allein stehende Komponente für die Groupware-Migration genutzt werden oder im Zusammenhang mit dem Web Server. Danach werden die Datei-, Netz- und Druckdienste migriert. Den Abschluss bildet die Umstellung des Desktops mit allen Fach- und Officeanwendungen.

Bei der sanften Migration können die Komponenten nicht beliebig ausgetauscht werden, sondern was zusammen gehört muss auch zusammen bleiben. Wichtig ist auch, dass das Projekt zeitlich nicht überstrapaziert und ein realistischer Endtermin gesetzt wird. Der administrative Aufwand an die IT-Umgebung ist höher als bei der schnellen Migration, aufgrund der Doppelgleisigkeit der IT-Systeme. Letztendlich sollte auch bei einer

sanften Migration die Umstellung in zwei bis drei Phasen durchgeführt werden, um erfolgreich abgeschlossen werden zu können.

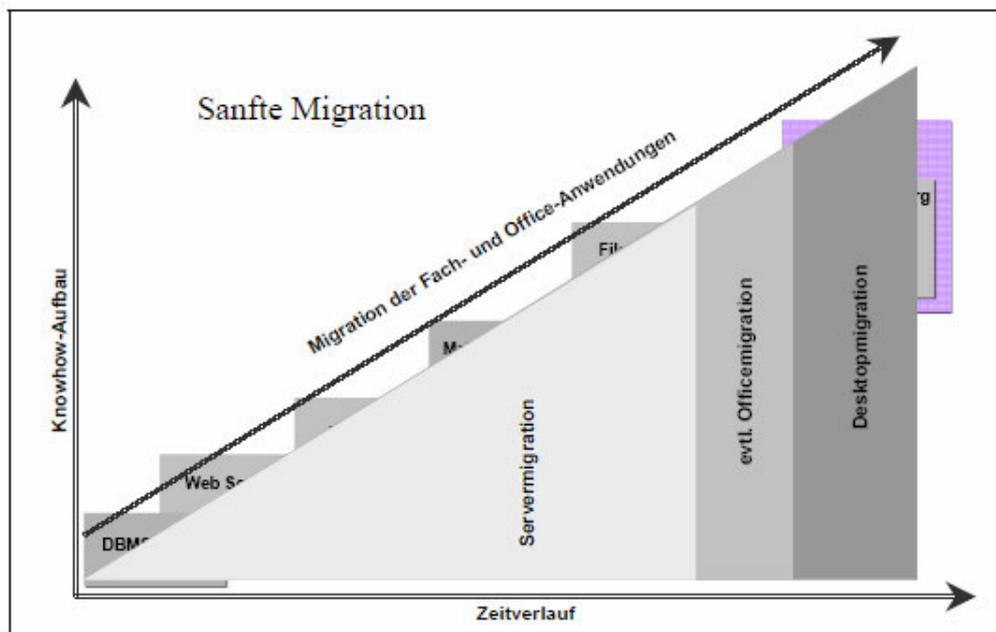


Abbildung 14: Phasen der Umstellung bei einer sanften Migration (aus: Bundesministerium des Innern - BMI (2003), S. 387)

Das obige Bild zeigt die drei Phasen der sanften Migration. Aufgrund der Kompatibilität und technischen Reife mancher OSS-Produkte kann die Umstellung auch in einem heterogenen System weit vorangetrieben werden. Erst am Ende, wenn alle Anwendungen umgestellt sind, kann die Desktop-Migration vorgenommen werden. Es kann sich mitunter auch bewähren, dass man frühzeitig auf OSS-Produkte umstellt, die auch für Windows verfügbar sind (z.B: OpenOffice.org), um den Benutzer an die neue Umgebung zu gewöhnen.<sup>104</sup>

<sup>104</sup> Bundesministerium des Innern – BMI (2003), S. 385 ff

### 5.3.3 Kritische Erfolgsfaktoren

Egal welcher Migrationsprozess<sup>105</sup> gewählt wird, das Projekt an sich ist komplex und vielschichtig. Die nachfolgende Abbildung verdeutlicht die vielen Phasen des Migrationsprozesses.

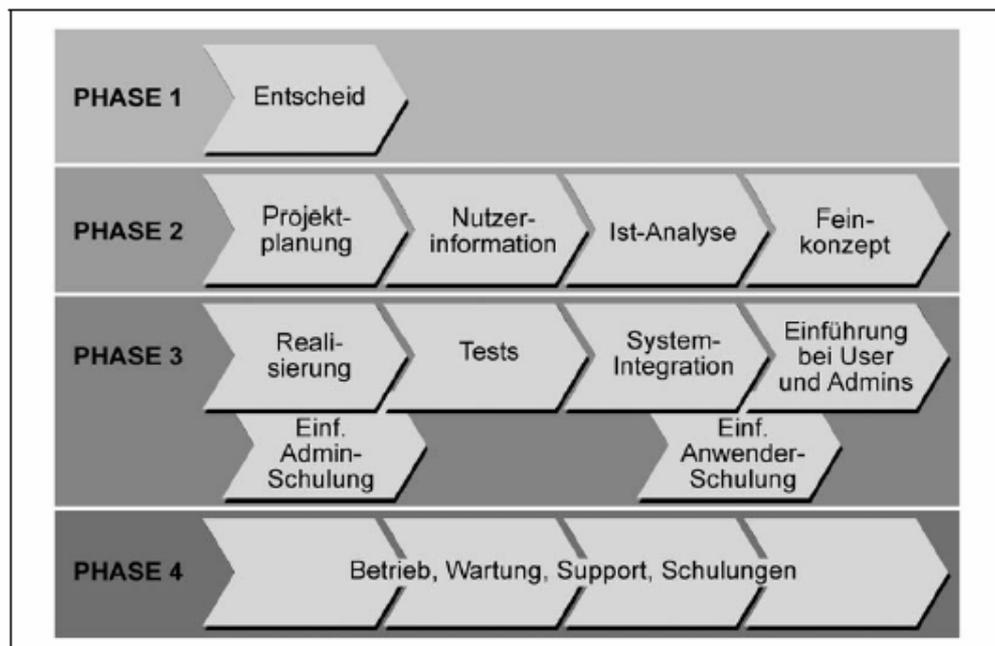


Abbildung 15: Modell: stufenförmiger Migrationsprozess (aus: Bundesministerium des Innern - BMI (2003), S. 388)

Um Migrationsprojekte zu einem erfolgreichen Abschluss zu bringen, gilt es die kritischen Erfolgsfaktoren am Beginn zu definieren und zu bewerten. Erfolgreich ist ein Projekt dann, wenn das gewünschte Ziel innerhalb des geplanten und vereinbarten Zeit- und Budgetrahmens liegt. Zusätzlich sind „weiche“ Faktoren (z.B. Mitarbeiterzufriedenheit, Frust, Doppelarbeiten, Akzeptanz der neuen IT-Landschaft) nicht zu vernachlässigen, da auch sie einen Beitrag zum Gesamterfolg bringen.

Unabhängig von der Unternehmensgröße und von der Migrationsweise tragen die nachfolgenden Parameter zum Projekterfolg bei:<sup>106</sup>

- Formulierung eindeutiger Ziele des Migrationsprojektes: strategische Managementziele und Ziele auf Ebene der Servermigration

<sup>105</sup> komplette, teilweise oder punktuelle Migration

- Einbindung und Positionierung der Leitungs- und Entscheidungsebene: Informationen zur Verfügung stellen für Schlüsselentscheidungen für das Migrationsprojekt
- frühe Informationen und Einbindung der Zielgruppen/Mitarbeiter: Allen Beteiligten und Beschäftigten die geplanten Vorhaben transparent präsentieren
- Schaffung einer hohen Nutzerakzeptanz hinsichtlich der Zielumgebung: Kommunikation des Nutzens zu den Anwendern, um eine hohe Akzeptanz sicherzustellen
- Strukturierte Zeit-, Projekt- und Ressourcenplanung mit Projektcontrolling: Erstellung der Meilensteine und Ermittlung der Aufwände mit Überwachung dieser
- Organisatorische Maßnahmen zur Vorbereitung der Migration und Bildung eines qualifizierten Projektteams: eigene Projektorganisation definieren
- Detaillierte Bestandsaufnahme mit Definition der funktionalen Anforderungen: Bedarf und Anforderungen an die zukünftige IT-Umgebung ermitteln
- optimale Produkt- und Dienstleistungsauswahl: Wirtschaftlichkeitsanalyse der künftigen IT-Infrastruktur
- zeitnahe und nachhaltige Schulungen: Für Administratoren und Benutzer zeitlich optimiert
- Qualitätsmanagement und Dokumentation: Um alle Arbeitsschritte für Dritte nachvollziehbar zu machen und um die Qualität des Systems zu sichern.

Aus den oben genannten Erfolgsfaktoren erkennt man, dass das Migrationsprojekt mit dem Kauf oder Implementierung der entsprechenden Komponenten nicht abgeschlossen ist. Beginnend mit den Vorarbeiten, fortführend mit dem eigentlichen Migrationsprozess bis hin zur Nachbearbeitung sind weitergehende Aktivitäten und Abhängigkeiten zu berücksichtigen.

---

<sup>106</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 387 ff

Es gilt, dass Migrationsprojekte nur dann erfolgreich und wirtschaftlich sind, wenn neben der Minimierung der laufenden Kosten ebenfalls eine Verbesserung der Aufgabenbearbeitung durch eine moderne und funktional zielgerichtete IT-Infrastruktur erreicht wird. Im Allgemeinen soll eine Erhöhung der Flexibilität, Leistungsfähigkeit und Reaktionsbereitschaft bei vorhandenen und zukünftigen Aufgabenstellungen erreicht werden. Weitere Ziele wie die Hersteller- und / oder Plattformunabhängigkeit sind wichtige Aspekte, die einer langfristigen Wirtschaftlichkeitsbetrachtung standhalten müssen.<sup>107</sup>

Eine detaillierte Beschreibung der einzelnen Erfolgsfaktoren würde den Umfang dieser Thesis sprengen. Für tiefgehendere Informationen diesbezüglich wird auf die verfügbare Literatur wie der Migrationsleitfaden der KBSt (BMI) verwiesen.

## **5.4 Migrationsarten**

### **5.4.1 Grundlegendes**

Von der Wirtschaftlichkeitsbetrachtung ausgehend, kann man folgende Migrationsarten aufzeigen:

- Vollständige Migration
- Fortführende Migration
- Teilmigration (punktuell und serverseitig)

Um die Berechnungen nicht zu verkomplizieren, wurde der Faktor Hardware bei der Betrachtung der Migrationskosten außer Acht gelassen. Wenn die Hardwareausstattung nicht älter als 2-3 Jahre ist, ist eine Migration ohne Veränderung möglich. Ist die Hardware älter, so kann eine Erneuerung derselben erforderlich werden, unabhängig von der Migrationsrichtung (Open Source oder proprietäre Software).

---

<sup>107</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 387 ff

Bei der monetären Betrachtung wird ein typischer Nutzungszeitraum von fünf Jahren angenommen. Eine Reinvestition wird erst nach diesem Zeitpunkt erwartet. Die gilt sowohl für eine Migration in Richtung OSS als auch in Richtung proprietärer Software. Folgekosten in den vier Jahren nach der Beschaffung werden nicht angenommen. Im Wesentlichen werden einmalige Beschaffungspreise in die Berechnung eingerechnet. In die Berechnung werden auch Mietvarianten (ASP) integriert.

Ein Vergleich der benutzerbezogenen Migrationskosten für die vollständige und fortführende Migration zeigt Kostenvorteile für die OSS-Umgebung (=vollständige Migration):<sup>108</sup>

<b>Unternehmenstyp</b>	<b>Vollständige Migration</b>	<b>Fortführende Migration</b>
Klein	533 €	862 €
Mittel	365 €	830 €
Groß	323 €	821 €

Tabelle 4: Vergleich der benutzerbezogenen Migrationskosten für vollständige/fortführende Migration

Vergleicht man nun die benutzerbezogenen Migrationskosten bei der OSS- und CSS Migration, so liegen die Kosten für die proprietäre Software um einiges höher. Der Kostenvorteil für OSS liegt vor allem in den eingesparten Lizenzkosten, die bei fast 98 Prozent liegen. Beim Vergleich der Personalkosten liegen die Kosten für OSS zwischen 12 und ca. 55 Prozent höher.

---

<sup>108</sup> Die genauen Berechnungen sind im Anhang zu finden

Der Trend zeigt bei den Migrationsformen eine annähernd gleiche Entwicklung – die Kosten steigen, je kleiner die Organisation ist.<sup>109</sup>

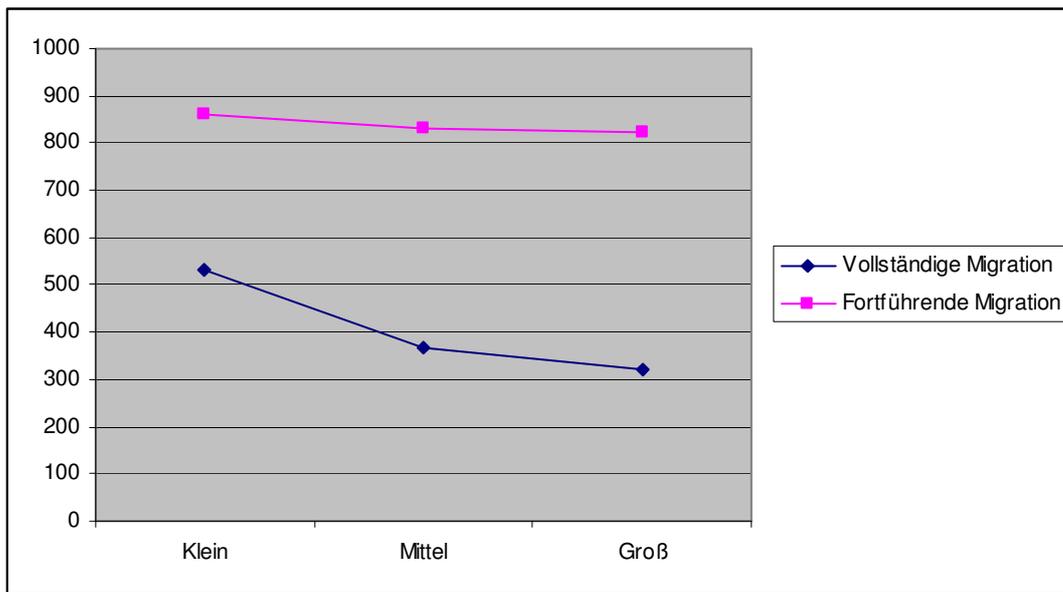


Abbildung 16: Migrationskostenentwicklung

### 5.4.2 Vollständige Migration

Wie Beispielrechnungen zeigen, dominiert der Personalkostenanteil, welcher bei ca. 90% liegt. Für die unterschiedlichen Unternehmenstypen ergibt sich folgende prozentuale Verteilungsbandbreite der Migrationskosten.<sup>110</sup>

Unternehmenstyp	Software	Personal
Klein	Bis zu 2%	Bis zu 98%
Mittel	Bis zu 1%	Bis zu 99%
Groß	Bis zu 1%	Bis zu 99%

Tabelle 5: Verteilung der Kosten bei "Vollständiger Migration" in Behörden

<sup>109</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 306 ff, Anhang

<sup>110</sup> Die genauen Berechnungen sind im Anhang zu finden

Die in diesem Modell gerechneten Einsparungen sind die jeweils zu tätigen Aufwendungen für eine Migration in die Windows 2003 Umgebung.<sup>111</sup>

<b>Unternehmenstyp</b>	<b>Basis Beschaffung (Einmalpreis)</b>
Klein	533 €
Mittel	365 €
Groß	323 €

Tabelle 6: Gesamt-Migrationskosten je User bei vollständiger Migration

Die Migration auf eine OSS-basierte Umgebung bedeutet insbesondere bei großen Unternehmen einen überdurchschnittlichen Einspareffekt. Die Einsparung bei der Migration von Windows NT auf Linux im Gegensatz zu Windows 2003 führt zu Ersparnissen, die fast das Dreifache der Aufwendungen übersteigen.

Die Personalkosten bei der Umstellung liegen in ähnlichen Größenordnungen, so dass der größte Anteil der Einsparungen aus nicht benötigten Lizenzkosten stammt.

Bei mittleren und kleinen Unternehmen zeigt sich ein vergleichbares Szenario.

Daraus folgt, dass eine Migration in die Open Source Umgebung sehr empfehlenswert ist.<sup>112</sup>

---

<sup>111</sup> Die genauen Berechnungen sind im Anhang zu finden

<sup>112</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 308 f

### 5.4.3 Fortführende Migration

Bei dieser Migrationsart können keine Ersparnisse erkannt und gegenge-rechnet werden. Daher erfolgt eine alleinige Darstellung der Szenario-bezogenen Kostenvolumina.<sup>113</sup>

<b>Unternehmenstyp</b>	<b>Basis Beschaffung</b>	<b>Mietmodell</b>
Klein	862 €	948 €
Mittel	830 €	1.788 €
Groß	821 €	1.788 €

Tabelle 7: Gesamt-Migrationskosten je User bei fortführender Migration

Die Migration wird mit steigender Anwenderzahl rentabler. Ein Umstieg von Beschaffungspreisen auf Mietpreise lohnt sich nach dem zugrunde gelegten Modell nicht.<sup>114</sup>

### 5.4.4 Teilweise Migration

#### 5.4.4.1 Punktuelle Migration

Diese Migrationsform beinhaltet die dauerhafte Ablösung einer ausgewähl-ten Systemkomponente innerhalb der gesamten IT-Struktur. Grundsätz-lich hat diese Migrationsform messbare Kostenvorteile gegenüber einer rein Microsoft-internen Migration (= fortführende Migration).<sup>115</sup>

<b>Unternehmenstyp</b>	<b>Basis Beschaffung (Einmalpreis)</b>
Klein	81 €
Mittel	45 €
Groß	31 €

Tabelle 8: Gesamt-Migrationskosten je Benutzer bei punktuellerer Migration

---

<sup>113</sup> Die genauen Berechnungen sind im Anhang zu finden

<sup>114</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 309 f

Die Kostenaufteilung von Software und Personal bewegt sich um die 90% Marke. Je nach personeller Intensität der Umstellungsprozesse kann dieser Anteil variieren. Es wird für die weitere Berechnung angenommen, dass sowohl für OSS ein Personalkostenanteil von ca. 90% realistisch ist.<sup>116</sup>

<b>Unternehmenstyp</b>	<b>Software</b>	<b>Personal</b>
Klein	Bis zu 10%	Bis zu 90%
Mittel	Bis zu 8%	Bis zu 92%
Groß	Bis zu 6%	Bis zu 94%

Tabelle 9: Migrationskostenverteilung

#### 5.4.4.2 Serverseitige Teilmigration

Diese Migrationsform entspricht dem serverseitigen Teil der vollständigen Migration.

Alternativ zur vollständigen Migration wird mit dieser Form der Umstellung ein relativ hoher Wirkungsgrad bezogen auf Dringlichkeits- und Qualitäts-/Strategiekriterien erzielt, da die Migration unter den Gesichtspunkten des Serverbereichs liegt.

Diese Migrationsalternative bietet sich nicht nur aus Kostengründen an, sondern sichert auch einen sanften, für die Anwender kaum spürbaren Übergang in die OSS-Welt.<sup>117</sup>

<b>Unternehmenstyp</b>	<b>Basis Beschaffung (Einmalpreis)</b>
Klein	296
Mittel	269
Groß	248

Tabelle 10: Gesamt-Migrationskosten je Benutzer bei serverseitiger Teilmigration

<sup>115</sup> Die genauen Berechnungen sind im Anhang zu finden

<sup>116</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 310

<sup>117</sup> Die genauen Berechnungen sind im Anhang zu finden

Die Migrationskosten betreffen hier zwar nur die Serverplattform, werden jedoch im Vergleich mit anderen Kosten auf die Anzahl der Benutzer der Organisation bezogen.<sup>118</sup>

<b>Unternehmenstyp</b>	<b>Vollständige Migration</b>	<b>Serverseitige Teilmigration</b>	<b>Client-Anteil in vollständiger Migration</b>
Klein	533	296	237
Mittel	365	269	96
Groß	323	248	75

Tabelle 11: Migrationskostenvergleich vollständige und serverseitige Migration

Die Kostenentwicklung bei der breiten Migration bestätigt den bisher festgestellten Trend der Kostendegression bei steigender Organisationsgröße. Im Vergleich zur vollständigen Migration zeigt sich ein relativ konstant verlaufender Anteil für die clientseitigen Umstellungskosten.<sup>119</sup>

## **5.5. Qualitativ-strategische Betrachtung**

### **5.5.1 Grundsätzliche Betrachtung**

Die Frage des zukünftigen Betriebssystems ist eine wichtige strategische Entscheidung. Diese Entscheidung kann nicht nur auf Grund der Wirtschaftsbetrachtung getroffen werden, da nicht alle Auswirkungen monetär abschätzt werden können. Daher ist eine qualitativ-strategische Betrachtung notwendig.

Die qualitativ-strategische Betrachtung analysiert die Auswirkungen, die sich entweder nicht oder nur sehr schwer in monetären Einheiten beschreiben lassen.

---

<sup>118</sup> Die genauen Berechnungen sind im Anhang zu finden

<sup>119</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 311 f

Im Folgenden wird die Nutzwertanalyse als Instrument der qualitativ-strategischen Betrachtung angeführt. Dadurch soll eine transparente Nachvollziehbarkeit der Ergebnisse erreicht werden.<sup>120</sup>

### 5.5.2 Nutzwertanalyse

Auf Basis der IT-WiBe wird ein Kriterienkatalog präsentiert, der als Grundlage für unternehmensspezifische Untersuchungen gelten soll. Dabei soll nicht die Vollständigkeit im Vordergrund stehen, da für jedes Unternehmen der Kriterienkatalog im speziellen angepasst werden muss. In erster Linie soll der Kriterienkatalog einen Überblick über die qualitativ-strategischen Gesichtspunkte einer Migration geben. Für die Beschreibung der Punktebewertung je Kriterium soll die Notenskala von 0 bis 10 verwendet werden, wobei 10 der Maximalwert ist. Die Punkte werden mit der jeweiligen Gewichtung multipliziert und je Kriterium summiert. Ein Wert größer als 500 stuft das IT-Vorhaben als „empfehlenswert zur Durchführung“ ein.<sup>121</sup>

	<b>Kriterium</b>	<b>Gewichtg.</b>	<b>Punkte</b>
1.	<i>Dringlichkeitsfaktoren</i>	100	
1.1	Ablösedringlichkeit Altsystem	20	
1.2	Fehler und Ausfallhäufigkeit im Altsystem	20	
1.3	Wartungsprobleme, Personalengpässe	20	
1.4	<i>Flexibilität Altsystem</i>		
1.4.1	Ausbau- / Erweiterungsgrenzen	10	
1.4.2	Interoperabilität, Schnittstellenprobleme (aktuell / zukünftig)	10	
1.4.3	Benutzerfreundlichkeit	20	

<sup>120</sup> Vgl. Unilog Integrata Unternehmensberatung (2003), S. 20

<sup>121</sup> Vgl. Unilog Integrata Unternehmensberatung (2003), S. 20 ff, Bundesministerium des Innern – BMI (2003), S. 354 ff

2.	<i>Qualitativ-strategische Faktoren</i>	100	
2.1	Einpassung in den IT-Ausbau des Unternehmens gesamt	5	
2.2	Folgewirkung für Kommunikationspartner	15	
2.3	Herstellerunabhängigkeit	15	
2.4	Attraktivität der Arbeitsbedingungen	5	
2.5	Qualitätssicherung/ -erweiterung	2	
2.6	Verbreitung/Verfügbarkeit der Ausbildung	3	
2.7	Imageverbesserung	2	
2.8	<i>Verbreitung/Verfügbarkeit der Software</i>		
2.8.1	Marktdurchdringung	5	
2.8.2	Unabhängiger Support	5	
2.8.3	Vorhandene Zertifizierung der Software	5	
2.8.4	Verfügbare Admin-Tools für die Software	5	
2.9.	<i>IT-Sicherheit</i>		
2.9.1	Kommunikationssicherheit	6	
2.9.2	Applikationssicherheit	6	
2.9.3	Ausfallsicherheit	6	
2.9.4	Sicherheitsmanagement	6	
2.9.5	Investitions- und Planungssicherheit	9	

Tabelle 12: Kriterienkatalog für die qualitativ-strategische Betrachtung (aus: Bundesministerium des Innern - BMI (2003), S. 354)

Die Festlegung der Gewichtung der Dringlichkeit und der qualitativ-strategischen Betrachtung folgt im Unternehmen selbst. Empfohlen wird ein Verhältnis von 40:60 aus Sicht der Dringlichkeit.

Die Nutzwertanalyse muss für alle möglichen Migrationsszenarien durchgeführt werden. Als Endergebnis erhält man eine Tabelle, die an Hand der Gesamtpunktzahl die Lösung mit den größten Vorteilen ermittelt. Somit kann eine Reihung der Alternativen vorgenommen werden.

Die Erläuterung der einzelnen Kriterien muss aufgrund der unterschiedlichen Voraussetzungen bei der Migration im jeweiligen Unternehmen indi-

viduell festgelegt werden. Eine gute Anleitung hierzu, was das Kriterium umfassen kann, bietet der Migrationsleitfaden der KBSt (BMI) auf den Seiten 354 bis 359.

## **5.6. Migrationsempfehlungen**

### **5.6.1 Weg der Entscheidungsfindung**

Ausschlaggebend für eine Migrationsempfehlung sind die Ergebnisse der langfristig angelegten Wirtschaftlichkeitsanalyse. Auch wenn eine Migration aus technischer Sicht ohne Einschränkung möglich ist, so können wirtschaftliche Überlegungen ein anderes Ergebnis erzeugen. Aufgrund der Zusammenhänge der Komponenten bei IT-Systemen kann eine Entscheidungsfindung nur als langfristige Entscheidung betrachtet werden. Üblicherweise werden folgende Strategien in der Wirtschaft verfolgt:

- Auf Basis von offenen Standards und Spezifikationen eng aufeinander abgestimmte System- und Anwendungsplattformen, gegebenenfalls unter zusätzlichem Einsatz von spezialisierten Integrationsprodukten
- Auf Basis von herstellereigenen (nicht oder nur zum Teil offenen Schnittstellen und Spezifikationen), eng aufeinander abgestimmten System- und Anwendungsplattformen, ggf. unter Einsatz von herstellereigenen Integrationsprodukten (z.B: VMWare)
- Einsatz von Insel-Lösungen zur punktuellen Abdeckung von Fachverfahren und -anwendungen (historisch)
- Auf Basis von offenen Standards und Spezifikationen aufeinander abgestimmte System- und Anwendungsplattformen mit Nutzung des offenen (wieder verwendbaren) Source Codes (insbesondere bei OSS)

Während gewisse Entscheidungen zum punktuellen Umstieg auf OSS einfach zu fällen sind (z.B: Umstieg auf Apache Webserver) sind Entschei-

dungen zum flächendeckenden Einsatz viel umfangreicher und erfordern eine methodische Vorgehensweise. Deren elementare Meilensteine sind:

- Erarbeitung einer Gesamt IT-Strategie unter Berücksichtigung der bestehenden finanziellen, organisatorischen, innovationsbedingten und personellen Zielsetzungen
- Definition der künftigen Open Source Plattform-Strategie unter Berücksichtigung der langfristigen Wirtschaftlichkeitsberechnung im Hinblick auf den Einsatz von freien und kommerziellen Standardprodukten
- Festlegung aller zur Sicherstellung der internen und externen Wiederverwendbarkeit sowie Interoperabilität notwendigen Standards in einem Blueprint Katalog
- Auswahl der Produkte zur Abdeckung der Anforderungen
- Definition des Vorhabens mit dem dazugehörigen Zeit- und Aktionsplan, sowie Sicherstellung einer Budgetierung <sup>122</sup>

Die einzelnen Phasen können mit gängigen Methoden und Werkzeugen bewerkstelligt werden.

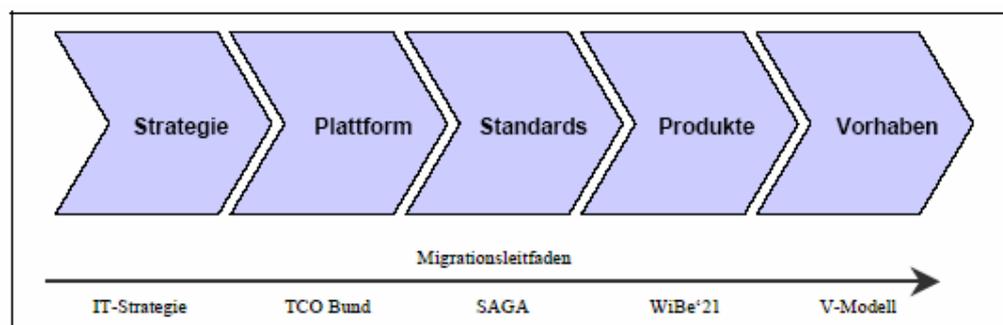


Abbildung 17: Entscheidungsprozess zur Einführung von OSS (aus: Bundesministerium des Innern - BMI (2003), S. 361)

## 5.6.2 Grundsatzempfehlungen

Allgemeingültige Empfehlungen können aufgrund der unterschiedlichen Ausgangssituationen in den Unternehmen nur selten getroffen werden.

<sup>122</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 360 f

Es gilt, dass mit wachsendem Grad der Integration der Produkte einer Plattform die Wirtschaftlichkeit insgesamt aus mehreren Gründen zunimmt:

- durch höhere Produktivität, bei gut (ohne Systembrüche) aufeinander abgestimmten Produkten
- durch die wachsende Wiederverwendbarkeit von Komponenten und Lösungen, die mit gleicher Middleware-Technologie entwickelt wurden
- durch Einsparungen bei Vereinheitlichung von Beschaffungs- und Wartungsprozessen und gegebenenfalls -verträgen

Darüber hinaus gilt, dass mit wachsendem Grad der Standardisierung auf Basis von offenen Standards die Wirtschaftlichkeit aus mehreren Gründen zunimmt:<sup>123</sup>

- durch den einsetzenden Wettbewerb von Produkten und Lösungen
- durch eine geringere Herstellerabhängigkeit
- durch einen insgesamt breiteren Dienstleistungsmarkt

Davon ausgehend können folgende Grundsatzempfehlungen zum Einsatz von OSS Produkten aufgestellt werden:

- Empfehlung für die Verankerung der Wirtschaftlichkeit als Leitbild der Gesamt IT-Strategie bei angemessener Berücksichtigung der Faktoren Innovation und Organisation
- Empfehlung für den Einsatz des Betriebssystems Linux als Grundlage der IT-Plattform für alle Anwendungsbereiche, falls die Voraussetzungen für eine Voll- oder Teilmigration zutreffen
- Empfehlung für den Einsatz von offenen, von IT-Industrie und Open Source Community gleichermaßen anerkannten Standards als Grundlage zur Auswahl und Integration von SW-Produkten zur Vermeidung extremer Herstellerabhängigkeiten

---

<sup>123</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 361 f

- Empfehlung zur Durchführung einer projektbezogenen Wirtschaftlichkeitsbetrachtung im Entscheidungsprozess für den Einsatz offener und kommerzieller Linux-Produkte

Grundsätzlich kann eine Umstellung auf die OSS-Plattform die ökonomisch sinnvollere (rentablere) Variante gegenüber einer fortführenden Migration auf neue Microsoft-Versionen sein. Der Wegfall bzw. die Reduktion der Lizenzkosten kann in mehreren Fällen zu monetären Einsparungen führen, wie:

- Serverseitige Teilmigration, verbunden mit einer HW- und SW-Konsolidierung, wenn Unix Know-how und Unix-Systeme bereits vorhanden sind
- Punktuelle Ablösung von Mitgliedern der ehemaligen MS Back-Office Familie (heute .NET Enterprise Server), beispielsweise Exchange oder SQL Server, insbesondere mit größeren oder wachsenden Nutzer- und somit Lizenzzahlen
- Clientseitige Teilmigration von MS Office Produkten, wenn die Nutzung von Office als Laufzeitumgebung für Makros oder Anwendungen die Ablösung nicht verhindert

In vielen Einsatzszenarien muss zur Beurteilung der Einsparmöglichkeiten die strategische Dimension herangezogen werden. Es fallen Schulungskosten an, egal auf welche Zielplattform umgestellt wird (OSS bzw. Windows). Daher ist dieser Kostenblock als neutral für den direkten Vergleich zu sehen. In jedem Fall sind Umstellungskosten der gegebenenfalls existierenden Fachanwendungen zu berücksichtigen.

Da die Grundsatzempfehlung nicht alle Anforderungen und Rahmenbedingungen der jeweiligen Ausgangssituationen berücksichtigen können, folgen in den folgenden Abschnitten die detaillierteren Empfehlungen für die unterschiedlichen Szenarien.<sup>124</sup>

---

<sup>124</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 362 f

### 5.6.3 Vollständig „ablösende“ Migration

Die vollständige Migration setzt nur OSS Komponenten in allen Ebenen der IT-Infrastruktur ein. Hierbei stehen grundsätzlich zwei Varianten der Software zur Verfügung, die auch kombiniert werden können:

- OSS: Open Source Software (oder freie Software): quelloffene und kostenlose Software, entwickelt durch die OSS-Community
- COLS: COmmercial Linux Software: kommerzielle quelloffene oder proprietäre Software für Linux, als Angebot der SW-Hersteller

Die in vielen Unternehmensbereichen benutzten ERP-basierten Anwendungssysteme und / oder eigenentwickelten und auf Windows basierenden Applikationen kann keine vollständige Abdeckung mit OSS erreicht werden. Aufgrund der derzeit positiven Förderung von Linux und die Verfügbarkeit von umfangreichen Applikationen wie SAP oder Oracle ist der Einsatz von OSS und COLS grundsätzlich als positiv zu beurteilen.

Die individuelle Ausprägung der möglichen und empfohlenen Systemarchitekturen ergibt sich aus der Größe, der IT-Intensität („Lastigkeit“) und dem Spezialisierungsgrad der Unternehmung. Hierbei spielen einerseits die Skalierbarkeit und Verfügbarkeit der einzelnen Komponenten und andererseits der Aufwand für die Einführung eine Rolle.

Daraus folgt, dass im Folgenden die Schwerpunkte für die jeweiligen Unternehmensgrößen betrachtet werden.<sup>125</sup>

### 5.6.3.1 Architekturmodell

Am Beginn folgt ein allgemeines Architekturmodell, welches für alle Unternehmensgrößen anwendbar ist. Dabei wird von einem „Fat Client“ ausgegangen, d.h. alle Applikation sind lokal installiert.

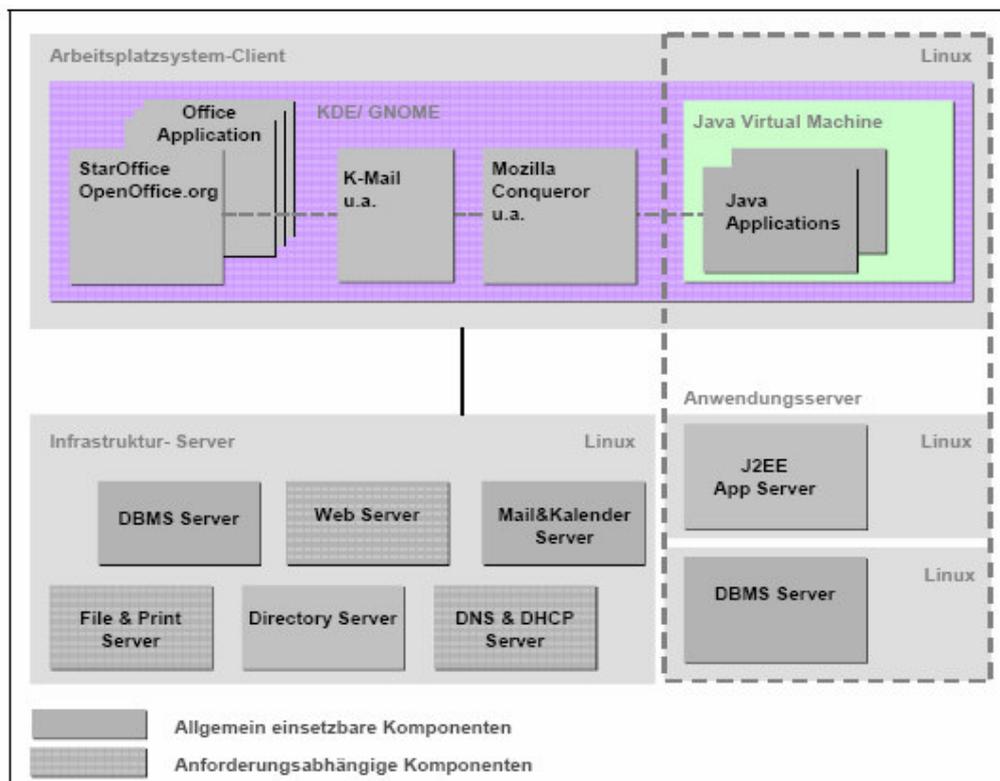


Abbildung 18: Allgemeine Systemarchitektur mit einem linuxbasierten Fat Client

(aus: Bundesministerium des Innern - BMI (2003), S. 365)

Die oben angeführte Konfiguration ist stellvertretend für einen multifunktionsfähigen Arbeitsplatzrechner mit einem handelsüblichen PC. Die Server-Plattform deckt die üblichen Infrastrukturaufgaben ab, darüber hinaus vervollständigt ein Anwendungsserver in der 3-schichtigen Architektur das Bild. Die ausgewählten Komponenten decken u.a. folgende Aufgabenbereiche ab:

- Arbeitsplatzrechner (Desktop und Office): Erfolgt auf Linux-Basis, wobei keine Distribution im Speziellen empfohlen werden kann. Jede

<sup>125</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 363 f

Distribution beinhaltet die wichtigsten Applikationen für eine möglichst breite Anwendungspalette.

- Groupware (Mail & Kalender Server): Die Firma Novell bietet ein Groupwareprogramm – Ximian – welches auch eine Verbindung zu einem MS Exchange Server aufbauen kann.
- Datenbanksysteme (DBMS Server): Die Firma MySQL bietet mit dem gleichnamigen Produkt eine gute Variante an. Für spezielle Anwendungen kann auf weitere, frei verfügbare Datenbanksysteme, wie MAX DB oder PostgreSQL, zurückgegriffen werden.
- Webserver: Hier wird Apache eingesetzt, welcher weltweit den größten Marktanteil hat.
- Dateiablage (File Server): Als Dateiservice wird das aus der UNIX-Welt bekannte und bewährte Network File System (NFS) verwendet. Die physikalische Speicherung wird mit einem Dateisystem mit Journaling-Eigenschaften verwirklicht, wie XFS oder EXT3.
- Druckdienste (Print-Server): Hierbei wird das Common UNIX Printing System (CUPS) empfohlen, welches allen großen Distribution beigelegt ist.
- Authentisierungsdienste: Auch hierfür gibt es geeignete Linux-Programme wie OpenLDAP.
- Netzwerkdienste (u.a. DNS & DHCP Server): Auch diese Anwendungen sind standardmäßig bei allen großen Distributionen vorhanden.
- Verzeichnisdienste: Sie spielen eine zentrale Rolle für die Sicherstellung der Effizienz des Systemmanagements und der IT-Sicherheit auf Basis des LDAP-Protokolls.
- Systemmanagementtools: Ein mögliches Produkt ist OpenView und Dienste, welche mit den Distributionen mitgeliefert werden.

Die staffierten Bereiche in obiger Abbildung können unabhängig von der Größe des Unternehmens eingesetzt werden.<sup>126</sup>

---

<sup>126</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 364 f

Diese Thesis soll keine speziellen Produktempfehlungen geben, da die Produkte laufend weiterentwickelt werden und somit ständig an Funktionalität zunehmen. Daher wird auf eine genauere Beschreibung der jeweiligen Anwendung verzichtet.

Die technischen Betrachtungen zeigen, dass bis auf wenige Ausnahmen für alle Microsoft Produkte OSS Alternativen vorhanden sind. Kritische Punkte bei der vollständigen Migration sind:

- Die Kompatibilität zwischen OpenOffice.org/StarOffice und MS Office ist nicht vollständig gegeben. Dies hat insbesondere Auswirkungen für jene Anwender, die häufig mit anderen Anwendern gemeinsame Dokumente erstellen müssen. Kommen in diesen Fällen beide Office-Varianten zum Einsatz, führt dies in der Regel zu Problemen bei der Formatierung.
- Die Chart-Engine von OpenOffice.org bzw. StarOffice weist nicht die gleiche Mächtigkeit auf wie die MS Excel Chart-Engine. Dies betrifft insbesondere die Erstellung von Charts auf Basis von Pivot-Tabellen.
- Zu einigen Produkten wie MS-Projekt oder Visio gibt es noch keine adäquate Alternative.

Es könnte auch noch andere wirtschaftliche Gründe geben, die gegen eine vollständige Migration sprechen. Dies ist insbesondere beim Desktop der Fall:

- MS Office: Der Umfang und die Komplexität der zu migrierenden Makros, Skripte, Vorlagen und Dokumente kann eine Migration nach OpenOffice.org oder StarOffice unwirtschaftlich machen.
- MS Access: Analoge Konvertierungsproblematik gilt für MS Access und die zu migrierenden Access-Anwendungen, die häufig zur Abdeckung einfacher Vorgangsautomatisierung benutzt werden.
- Fachanwendungen: Abhängig vom Grad der Nutzung von nativen Windows-Fachanwendungen kann im ungünstigen Fall eine ablösende Migration bis zur Verfügbarkeit von Alternativprodukten verhin-

dert werden. Dies gilt auch für Anwendungen, die auf Basis von MS Exchange erstellt wurden und es als Laufzeitsystem nutzen.

Für alle Unternehmen gilt, dass großer Wert auf die IT-Sicherheit, Leistungsfähigkeit und Skalierbarkeit gelegt wird.

Große Unternehmen haben spezielle Anforderungen das Systemmanagement zum Netzwerk- und Systemmonitoring. Replikationsmechanismen finden breite Verwendung, je mehr Benutzer verwaltet werden müssen.

Mittlere und kleine Unternehmen haben keine so hohen Anforderungen an die IT-Infrastruktur. Insbesondere bei kleinen Unternehmen werden Leistungen im IT-Bereich oftmals außer Haus erledigt oder im Mietmodell (ASP) eingekauft. Auch OSS Produkte können skaliert werden und somit die Anforderungen kleiner Unternehmen abdecken. Große Lösungen haben in der Regel einen Mehraufwand an Installation und Konfiguration.<sup>127</sup>

#### **5.6.4 Vollständig „fortführende“ Migration**

Die fortführende Migration bedeutet, dass die Migration mit Microsoft Produkten durchgeführt wird. Die Microsoft Produkte müssen nicht in allen Bereichen beibehalten werden, sondern können durch OSS-Produkte ersetzt werden.

Vor allem werden es wirtschaftliche Gründe sein, die eine fortführende Migration bedingen. Dies ist insbesondere dann der Fall, wenn das bestehende IT-System über einen sehr hohen Grad an Integration verfügt, die nur als Windows-Anwendungen verfügbar sind, wie:

- Anzahl der Fachverfahren, die nur als Windows-Anwendungen verfügbar sind
- Die Verfügbarkeit der Quellcodes dieser Fachverfahren
- Integrationstiefe der einzelnen Fachverfahren, insbesondere in die MS Office-Umgebung
- Der Umfang der Verwendung Microsoft-spezifischer

---

<sup>127</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 365 ff

- Entwicklungsumgebungen
- Schnittstellen
- Programmiersprachen
- Anzahl der MS Office spezifischen Makros und Scriptings (z.B. Implementation von abteilungsübergreifenden Workflows)

Der Aufwand für die zuvor beschriebene ablösende Migration steigt mit zunehmendem Grad der Integration. Eine Aussage, ab welchem Grad der Integration eine fortführende Migration bedingt, kann nur durch die Einzelbetrachtung der zu migrierenden Komponenten und durch das Ergebnis der Wirtschaftlichkeitsanalyse getroffen werden. Durch die zukünftige .NET-Plattform wird die Migration von Windows-Komponenten auf OSS-Komponenten erheblich erschwert. Nur durch rechtzeitige Teilmigration kann dieser Entwicklung vorgebeugt werden.<sup>128</sup>

Eine weitere Ausgangssituation kann sein, dass ein Unternehmen gerade eine fortführende Migration auf die neuesten Windows-Komponenten durchführt. Bezogen auf das in den Migrationsempfehlungen zugrunde liegende Architekturmodell kann im Maximalfall folgende Ausprägung vor der Migration vorliegen:

- Arbeitsplatzrechner: Windows 2000 Clients mit Office 2000
- Webserver: Internet Information Server 5
- Datenbankserver: MS SQL Server 2000
- Groupware/Messaging: Exchange 2000
- Verzeichnisdienst: Active Directory Service
- Infrastrukturdienste: Windows 2000 Server (Advanced Server) Dateiablage, Druckdienste und Netzwerkdienste

Sollten einzelne Komponenten noch nicht im Einsatz sein, kann im Rahmen der Teilmigration eine adäquate OSS-Lösung dafür vorgeschlagen werden. Aufgrund der Wirtschaftlichkeitsüberlegungen gilt, dass für be-

---

<sup>128</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 375 f

reits im Einsatz befindliche Komponenten keine ablösende Migration bzw. Teilmigration innerhalb der kommenden 4 – 5 Jahre empfehlenswert ist. Im Folgenden werden einige Aspekte aufgeführt, die dennoch als Migrationsempfehlung zu sehen sind. Den Unternehmen sollte gezeigt werden, wie man die Abhängigkeit von Microsoftprodukten minimieren kann. Ebenso sollen stichwortartig Empfehlungen gegeben werden, die eine spätere ablösende Migration erleichtern.

Es können folgenden Empfehlungen für eine Reduktion von der Abhängigkeit von Microsoftprodukten gegeben werden:

- Verzeichnisdienst: Auf die Nutzung von Active Directory sollte weitestgehend verzichtet werden und wenn überhaupt nur in minimaler Ausprägung.
- Desktop: Der Einsatz von MS Access-Anwendungen sollte vermieden werden. Außerdem sollten die aktuellen VBA-Anwendungen analysiert werden, ob eine Migration durchgeführt werden kann. Ebenso sollten Fachanwendungen, die MS Office Produkte voraussetzen sukzessive umgestellt werden.
- Dateiablage: Die aktuelle Konfiguration muss detailliert und vollständig dokumentiert werden. Ebenso sollte auf die Verwendung lokaler Gruppen verzichtet werden.
- Groupware/Messaging: MS Exchange Server sollte nicht als zentrales Mail-Programm zum Einsatz kommen. Weiters sollten in den öffentlichen Ordnern von Exchange keine Anwendungen eingesetzt werden.
- Webanwendungen: Aufgrund der Vielzahl an OSS-Produkten gibt es viele Alternativen zu Microsoftprodukten.
- System Management: Auch hierbei ist ein Einsatz von Produkten von Drittherstellern zu bevorzugen.
- Netzwerkdienste: OSS-Produkte sind in diesem Segment sehr ausgereift und haben oftmals die größten Marktanteile verglichen mit Microsoft Produkten.

- Middleware: Es soll auf eine Wiederverwendbarkeit geachtet werden, wenn Applikationen entwickelt werden. Insbesondere die neuen Technologien wie XML und Web-Services sollten zum Einsatz kommen.<sup>129</sup>

### **5.6.5 Punktuelle Teilmigration**

Die punktuelle Teilmigration ist die dauerhafte Ablösung einer ausgewählten Systemkomponente durch OSS-Produkte.

Eine mögliche punktuelle Migration besteht in der Ablösung von MS Office durch OpenOffice.org. Es sind dabei die genannten funktionalen Einschränkungen und insbesondere die wirtschaftlichen Konsequenzen zu berücksichtigen.

Die Durchführung einer punktuellen Migration bietet eine Reihe von Vorteilen:

- Übersichtlicher und gut planbarer Migrationsumfang. Notwendige Anpassungen halten sich in einem begrenzten Rahmen, was für Projektplanung und -steuerung einen essentiellen Vorteil bietet
- Die Möglichkeit, stufenweise Betriebskonzepte und Erfahrungen mit einer neuen Betriebssystemplattform aufzubauen
- Schulungsaufwand fällt nur für die jeweils fachlich involvierten Administratoren an. Die neu geschulten Administratoren können so auch als Multiplikatoren innerhalb der IT-Abteilung dienen

In Wirklichkeit ist die punktuelle Teilmigration eine häufig erprobte und daher bevorzugte Lösung. Die Besonderheit der Teilmigration wird im folgenden Kapitel anhand der serverseitigen Teilmigration behandelt.<sup>130</sup>

---

<sup>129</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 377 ff

<sup>130</sup> Bundesministerium des Innern – BMI (2003), S. 380 ff

### **5.6.6 Serverseitige Teilmigration**

Im Folgenden sollte ein empfehlenswertes Szenario für eine serverseitige Teilmigration vorgestellt werden. Als Ausgangssituation wird eine MS Windows Umgebung angenommen.

Im Prinzip gelten die Empfehlungen für eine vollständige Migration auch für die serverseitige Migration. Der einzige Unterschied ist, dass auf der Client-Seite weiterhin Windows-Komponenten verwendet werden.

Die zentrale Anforderung für eine serverseitige Migration besteht in der reibungslosen Zusammenarbeit zwischen linuxbasierten Serversystemen und windowsbasierten Clientsystemen.

Eine wichtige Anforderung ist die Ablösung der Dateiablage-, Druck-, Netzwerk- und Authentifizierungsdienste inkl. der Migration der vorhandenen Datei- und Rechtestrukturen und der Übernahme der Konfigurationsdaten. Näheres wurde bereits in den vorangegangenen Kapiteln beschrieben.

Für die Benutzerverwaltung wird der Samba-Server in Kombination mit OpenLDAP empfohlen. Noch dazu bietet Samba die uneingeschränkte Möglichkeit der Anbindung an einen Windows Client. Da Samba auch die Dateidienste beherrscht, steht einem Einsatz nichts mehr im Wege. Die Druckdienste können mit CUPS verwirklicht werden, da CUPS optimal in Samba integriert werden kann.

Im Rahmen der Wirtschaftlichkeitsbetrachtung zeigt sich, dass ca. 65% - 80% der Projektkosten als Ersparnis gegengerechnet werden können. Die darüber liegenden Mehraufwände resultieren insbesondere aus Mehraufwänden für Personal für Umstellungsarbeiten.

Für diese Migrationsart ist ein direkter monetärer Vorteil gegenüber der fortführenden Migration in der Regel nicht darstellbar. Es sind vielmehr die weichen Faktoren, die in Form der Kriterien für Dringlichkeit und Strategie, ausschlaggebend für die Projektentscheidung sind.<sup>131</sup>

### **5.6.7 Gesamtergebnis**

Wenn man die wirtschaftlichen Aspekte der Migrationsarten vergleicht, stellt sich heraus, dass eine serverseitige Teilmigration die meisten Vorteile bietet. Somit ist diese Art der Umstellung auf Open Source grundsätzlich zu empfehlen.

Wenn ein Unternehmen die Microsoft-Umgebung beibehalten will, so ist eine fortführende Migration empfehlenswert.

Die vollständige sowie punktuelle Migration stellen unter wirtschaftlichen Gesichtspunkten suboptimale Verfahren dar. Jedoch kann aufgrund spezieller Erfordernisse bzw. Entscheidungen diese Varianten das bevorzugte Verfahren sein. Aber nicht umstellbare Fachanwendungen können die Unternehmen weiterhin zwingen, Microsoft-Systeme zu betreiben.<sup>132</sup>

---

<sup>131</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 382 ff

<sup>132</sup> Vgl. Bundesministerium des Innern – BMI (2003), S. 312

## 6. Fazit

Open Source Software ist ein bedeutendes Thema in der Computerindustrie. Was lange Zeit nur Informatikern bekannt war, bekommt in den letzten Jahren verstärkt Zulauf von „normalen“ Computeranwendern. Dies wird auch dadurch gefördert, dass immer mehr Zeitschriften, Magazine und Zeitungen Artikel über OSS verfassen.

Fast wöchentlich werden neue Horrormeldungen über ausgenutzte Sicherheitslücken in proprietärer Software kommuniziert. Als Lösung wird der Umstieg auf u.a. Open Source Softwareprodukte empfohlen, was langsam zum Umdenken der Anwender führt. Nicht zuletzt sinkt der Marktanteil des MS Internet Explorers zum ersten Mal seit Bestehen dieser Software. Dies ist ein deutliches Zeichen, dass OSS beginnt, Anwender außerhalb des Informatikbereichs für sich zu begeistern.

Die anfängliche Zurückhaltung von Anwendern, Wirtschaft und Verwaltung bei der Verwendung von OSS ist weitgehend verschwunden. Dies wurde durch eine einfach zu bedienende grafische Oberfläche, von Installationshilfen und dem verstärkten Angebot an unterschiedlichen Applikationen ermöglicht.

Obwohl derzeit ein Art Hype um Open Source Software ausgebrochen ist, wird es meiner Meinung nach, nicht zu einem ähnlichen Platzen der Seifenblase wie zur Jahrtausendwende kommen, als viele Internetgeschäfte in Konkurs gegangen sind (= Dot-Com Krise). Bei OSS steht nicht der schnelle Gewinn im Vordergrund der Entwickler, sondern der pure Spaß am Programmieren. Einzig bei den Supportanbietern und Dienstleistern wird es zur Marktbereinigung kommen. Dies wird nicht gleich, sondern erst in ein paar Jahren passieren, wenn der Markt erste Sättigungerscheinungen zeigt. Aktuell sagen alle Prognosen einen weiterhin stark wachsenden Markt voraus.

Meiner Meinung nach, ist Open Source Software für den Einsatz bei Privatanwendern bereits jetzt bestens geeignet. Ausnahmen sind spezielle Anwendungen, für die keine entsprechende OSS verfügbar ist (z.B: grafisch anspruchsvolle Spiele). Für alltägliche Tätigkeiten wie Textverarbeitung und Tabellenkalkulation finden sich gleichwertige OSS-Produkte.

Bei Unternehmen sieht die Situation anders aus. Je nach Größe des Unternehmens werden unterschiedliche Anforderungen an die Software wie Skalierung, Ausfallssicherheit oder Administrierbarkeit gestellt. Dies erfordert eine andere Funktionalität der Software. Dies ist relativ umständlich zu ermitteln, deshalb sollte hierbei auf professionelle Dienstleister zurückgegriffen werden.

Jeder IT-Verantwortliche, der plant, von kommerzieller Software auf offene Software umzusteigen, sollte sich der Tatsache bewusst sein, dass damit Kosten verbunden und Hürden zu überwinden sein können. Dies trifft auch bei der fortführenden Migration zu.

Da Microsoft aktuell seinen Support für Windows NT langsam aber sicher einstellt, schauen sich viele Unternehmen nach einer anderen Lösung um. In erster Linie galt bisher, dass einfach auf eine neuere Version des Microsoft Betriebssystems Windows gewechselt wird.

Mit der Alternative Linux und dem Ziel die IT-Kosten zu minimieren, wird das Auswahlverfahren vielschichtiger und komplexer.

Mitunter wird die Alternative Linux auch missbraucht, um auf Microsoft Druck auszuüben und eine Senkung der Lizenzkosten zu bewirken. Dies ist aber, meiner Meinung nach, der falsche Weg im Hinblick auf den Einsatz von OSS-Produkten im Unternehmen.

Dennoch zeigen die Erfolge von Open Source im Infrastrukturbereich, z.B: als Internetserver, Druckserver oder im Intraneteinsatz, dass dieses Softwaremodell funktioniert und qualitativ hochwertige Software entsteht. Wobei hier zu vermerken ist, dass der Hauptkonkurrent eher das teure Unix ist als Windows.

Im Folgenden sollen ein paar Empfehlungen zu OSS erwähnt werden, was die Akzeptanz bei Benutzern fördern könnte.

- Bestes aus beiden Welten: Es sollten Koexistenzlösungen angestrebt werden und die besten Lösungen eingesetzt werden.
- Ausbildung mit OSS: Schon in der Schule sollte OSS eingesetzt werden, um die jungen Anwender bereits diese Alternative zu zeigen. Mitunter könnten auch die Bildungsbudgets entlastet werden.
- Vertrieb von OSS ausbauen: Es sollte nicht nur die Software als Knoppix-Version (von CD ausführbar) in diversen Computerzeitschriften verteilt werden, sondern auch bei neu verkauften Computern sollte die Möglichkeit der Wahl zwischen proprietärem und offenem Betriebssystem bestehen.
- Einheitliche Distributionen: die Distributionen unterscheiden sich teils stark voneinander, obwohl sie alle auf dem gleichen Linux-Kernel aufbauen. Dies kann dazu führen, dass gewisse Software nur auf einer Distribution lauffähig ist. Hier wäre ein einheitlicheres Auftreten erwünscht.
- Kommunizieren von Erfolgsgeschichten: Die beste Werbung ist die Mund-zu-Mund Propaganda. Wenn der Einsatz von Linux in der Stadtverwaltung München ein Erfolg wird und nachweislich die Kosten gesenkt werden konnten, dann wird der wahre Ansturm auf OSS erst beginnen. Es kann jedoch auch ein Misserfolg (wirtschaftlich gesehen) werden.

## Quellenverzeichnis

- Arpagaus, Jürg/Seelhofer, Martin/Brüngger, Martin (2003): Ist-Situation Open Source Software in der Bundesverwaltung, Bern.
- Auf der Maur, Rolf (2003): Ohne Urheberrecht kein Open Source anlässlich der Computerworld Konferenz Open Source'03, 10/2003, Vortragsfolien  
<http://www.computerworld.ch/konferenz/2003/opensource/speech/index.html> (Stand: 21.10.2004)
- Bahr, Rudolf/Reiländer, Ralf/Troles, Egon (2000): Open Source Software in der Bundesverwaltung, in: KBSt Brief Nr. 2/2000, S. 1 – 11.
- Bezroukov, Nikolai (1999): A Second Look at the Cathedral and the Bazaar, in: First Monday, Vol. 4/No. 12, December 1999,  
[http://www.firstmonday.dk/issues/issue4\\_12/bezroukov/index.html](http://www.firstmonday.dk/issues/issue4_12/bezroukov/index.html) (Stand: 09.04.2004)
- Berlecon Research (Hrg.) (2002): FLOSS - Free/Libre and Open Source Software: Survey and Study  
[http://www.berlecon.de/output/studien\\_fb.php?we\\_objectID=67](http://www.berlecon.de/output/studien_fb.php?we_objectID=67), Berlin.
- Bundesministerium des Innern – BMI (Hrg.) (2003/Bd. 57): Migrationsleitfaden – Leitfaden für die Migration der Basissoftware auf Server- und Arbeitsplatz-Systemen, Berlin.
- Bundesministerium für Wirtschaft und Technologie – BMWi/Innominate AG (Hrg.) (2001): Open-Source-Software. Ein Leitfaden für kleine und mittlere Unternehmen, Berlin.
- Computerworld (2003): Verbreiteter als vermutet, 21.11.2003, S. 3.
- Computerworld-Magazin (2003): Open Source, 4/2003.
- Computerworld-Magazin (2004): Informatik für KMU, 2/2004, S. 18-21.
- Delio, Michelle (2000): It'll Be an Open-Source World, in: wired.com,  
[www.wired.com/news/print/0,1294,38240,00.html](http://www.wired.com/news/print/0,1294,38240,00.html) (Stand: 09.04.2004).
- DerStandard.at (04.08.2004): Fast jeder nutzt Open-Source-Software,  
<http://derstandard.at> (Stand: 14.08.2004).
- DerStandard.at (08.08.2004): Verwirrende Vielfalt von Open-Source-Lizenzen, <http://derstandard.at> (Stand: 14.08.2004).
- Ettrich, Matthias (2004): Koordination und Kommunikation in Open-Source-Projekten in: Gehrung, Robert/Luterbeck, Bernd (Hrg.) Open Source Jahrbuch 2004 - Zwischen Softwareentwicklung und Gesellschaftsmodell, Berlin, S. 179 – 192.
- Free Software Foundation (02.09.2004): Various Licenses and Comments about Them, <http://www.fsf.org/licenses/license-list.html> (Stand: 10.09.2004)
- Free Software Foundation (15.06.2004): What is Copyleft?,  
<http://www.fsf.org/copyleft/copyleft.html> (Stand: 10.09.2004)
- Ghosh, Rishab/Prakash, Vipul Ved (2000): The Orbiten Free Software Survey, in: First Monday, Vol. 5/No. 7, July 2000

- [www.firstmonday.dk/issues/issue5\\_7/ghosh/index.html](http://www.firstmonday.dk/issues/issue5_7/ghosh/index.html) (Stand: 09.04.2004).
- GNU Projekt (29.07.2001): Kategorien freier und unfreier Software, [www.gnu.org/philosophy/categories.de.html](http://www.gnu.org/philosophy/categories.de.html) (Stand: 10.09.2004)
- Hissam, Scott/Weinstock, Charles/Plakosh, Daniel (2001): Perspectives on Open Source Software, CarnegieMellon Software Engineering Institute, Pittsburgh.
- Holthaus, Marcus (2003): Sicherheitschancen und -risiken von Open-Source-Software im Geschäftsumfeld anlässlich der Computerworld Konferenz Open Source'03, 10/2003, Vortragsfolien <http://www.computerworld.ch/konferenz/2003/opensource/speech/index.html> (Stand: 21.10.2004).
- Howe, Carl D. (2000): Open Source Cracks The Code, Forrester Research, [www.forrester.com/ER/Research/Report/Summary/0,1338,9851,FF.html](http://www.forrester.com/ER/Research/Report/Summary/0,1338,9851,FF.html) (Stand: 09.04.2004).
- HP (10.09.2004): HP & Open Source, Firmenhomepage, <http://opensource.hp.com/> (Stand 10.09.2004).
- IBM (10.09.2004): Linux at IBM Solutions, Firmenhomepage, [http://www-1.ibm.com/linux/va\\_4049.shtml](http://www-1.ibm.com/linux/va_4049.shtml) (Stand: 10.09.2004).
- Janko, Herbert (2002): Pricing Model für Application Service Provider (ASP), Fachhochschule Liechtenstein.
- Klemme, Dieter/Arpagaus, Jürg/Rosenthal, David (2004): OSS-Strategie der Bundesverwaltung, Bern.
- Knyphausen-Aufseß, Dodo von (2003): Open Source Software Development als Herausforderung an die Softwareindustrie? anlässlich der Computerworld Konferenz Open Source'03, 10/2003, Vortragsfolien <http://www.computerworld.ch/konferenz/2003/opensource/speech/index.html> (Stand: 21.10.2004).
- Leiteritz, Raphael (2002): Der kommerzielle Einsatz von Open Source Software und kommerzielle Open Source-Geschäftsmodelle, Technische Universität Berlin – Fachbereich Informatik.
- Mass, Wolfgang (2003): CIO-Querschnittsstudie "Die strategische Option Open Source Software", St. Gallen
- Mitchell, David et. al. (2003): Linux on the Desktop: The Whole Story, Gartner Group.
- MySQL (10.09.2004): Online Shop, <https://order.mysql.com/> (Stand: 10.09.2004).
- Opensource (10.09.2004): Open Source Initiative OSI <http://www.opensource.org> (Stand: 10.09.2004)
- Raymond, Eric (1998): The Cathedral and the Bazaar, in: First Monday <http://www.tuxedo.org/~esr/writings/cathedral-bazaar/> (Stand: 09.04.2004).
- Redman, B.; Kirwin, W.; Berg T. (1998): TCO, A Critical Tool for Managing IT, Gartner Group.
- Rotz, Bruno von (2003): Open Source Software im Praxiseinsatz – Erfahrungen und Empfehlungen anlässlich der Computerworld Konferenz Open Source'03, 10/2003, Vortragsfolien

- <http://www.computerworld.ch/konferenz/2003/opensource/speech/index.html> (Stand: 21.10.2004).
- Röthig, Peter (2001/Bd. 52): WiBe 21 – Empfehlung zur Durchführung von Wirtschaftlichkeitsbetrachtungen in der Bundesverwaltung, insbesondere beim Einsatz der IT, Bundesministerium des Innern, Berlin.
- Silver, Michael A. (2002): Linux Desktop TCO: An Overview, Gartner Group.
- Silver, Michael A. (2003a): Linux Desktop Migration Cost Model, Gartner Group.
- Silver, Michael A. (2003b): Linux Desktop Migration: Finding the Break-Even Point, Gartner Group.
- Sourceforge (10.09.2004): Welcome, <http://sourceforge.net/index.php> (Stand: 10.09.2004).
- Sun Microsystems (10.09.2004): Sun's Linux Offerings, Firmenhomepage, <http://www.sun.com/software/linux/> (Stand: 10.09.2004).
- Unilog Integrata Unternehmensberatung (2003): Projekt Client Studie der Landeshauptstadt München, München.
- WCM – Computerzeitung für Österreich (2004): Linuxwochen 2004, Nr. 212, Mai 2004.
- Wendt, Mario (2003): Kommerzielle Software versus Open Source Software, Kosten- und Nutzeraspekte anlässlich der Computerworld Konferenz Open Source'03, 10/2003, Vortragsfolien <http://www.computerworld.ch/konferenz/2003/opensource/speech/index.html> (Stand: 21.10.2004)
- Wichmann, Thorsten (2004): Linux- und Open-Source-Strategien für CIOs, Berlecon Research, Berlin.
- Widmer, Ursula (2004): Gutachten betreffend Rechtsfragen bei Beschaffung und Einsatz offener Software in der Schweizerischen Bundesverwaltung (Projekt OPUS), Bern.
- Wieland, Thomas (2004): Stärken und Schwächen freier und Open-Source-Software im Unternehmen, in: Gehrung, Robert/Luterbeck, Bernd (Hrg.) Open Source Jahrbuch 2004 - Zwischen Softwareentwicklung und Gesellschaftsmodell, Berlin. S. 107 – 120.
- Zerdick, Axel/Picot, Arnold/Schrape, Klaus et.al. (1999): Die Internet-Ökonomie. Strategien für die digitale Wirtschaft, Springer.

## **Eidesstattliche Erklärung**

# **Open Source Software vs. Closed Source Software**

Wie brauchbar sind die Open Source Softwareprogramme  
für den Einsatz in Unternehmen?

Ich versichere, dass ich die vorstehende Arbeit selbstständig und ohne fremde Hilfe angefertigt und mich anderer als in der Arbeit angegebenen Hilfsmitteln nicht bedient habe.

Alle Stellen, die wörtlich oder sinngemäss aus Veröffentlichungen entnommen wurden, sind als solche kenntlich gemacht.

Wien, den 17. September 2004

Unterschrift:

(Herbert Janko)

# Anhang

## Generelle Annahmen für die Migrationsarten –

### Kapitel 5.4

Folgende Kosten/Nutzentreiber von Migrationsobjekten werden betrachtet:

- Hardwarekosten
- Softwarekosten
- Personalkosten für Datenübernahme

Andere Kostenarten werden neutral gesehen, wie:

- Kompatibilitätskosten
- Schulungskosten
- Administrationskosten

Für die Betrachtung der Einsparpotentiale werden bei der Migration von Windows auf OSS die nicht benötigten Lizenzkosten einer fortführenden Migration angenommen.

Folgende Personentage werden als Aufwand angenommen:

	<b>Klein</b>	<b>Mittel</b>	<b>Groß</b>
Fortführende M.	40	80	175
Vollständige M.	64	110	195
Punktuelle Teilmigration	9	13	18
Serverseitige Teilmigration	35	80	150

Diese Werte wurden dem Migrationsleitfaden der KBSt (BMI) Seite 316 ff entnommen und leicht abgeändert, da diese Thesis andere Organisationsgrößen definiert hat.

Bei externen Personentagen wird ein durchschnittlicher Tagessatz von 1000.- Euro angenommen.

Bei internen Personentagen wird ein durchschnittlicher Stundensatz von 40.- Euro angenommen (Tag: 8 Stunden -> 320,- Euro)

Die beispielhaften Personentage für große, mittlere und kleine Unternehmen werden jeweils im Verhältnis von 80/20 auf externe und interne Ressourcen aufgeteilt.

Für die Abzinsung künftiger Ersparnisse wird ein Zinssatz von 5 Prozent angenommen. Die in den Beispielrechnungen enthaltenen Preise erzeugen einen Liquiditätsabfluss im ersten Jahr. Da mit Beschaffungspreisen gerechnet wird, entstehen die Ersparnisse ebenfalls nur im ersten Jahr. Somit beträgt der Abzinsungsfaktor:  $1 / (1+0,05)^1 = 0,9524$

Die Softwarelizenzkosten betragen ca. 50% der normalen Listenpreise. Diese Konditionen sollten Unternehmen erreichen können.

Folgendes Szenario beinhaltet die Ermittlung der benötigten Lizenzsoftware, was wiederum das Einsparpotential für OSS ist:

<b>Kategorie</b>	<b>Anwenderzahl</b>	<b>Zielszenario</b>
Klein	Bis 100	<ul style="list-style-type: none"> <li>• eine Windows/Samba Domäne</li> <li>• ein Office Produkt je User</li> <li>• zwei Datei- /Druckserver</li> <li>• ein DBMS</li> </ul>
Mittel	101 bis 500	<ul style="list-style-type: none"> <li>• zwei Windows/Samba Domäne</li> <li>• ein Office Produkt je User</li> <li>• vier Datei- /Druckserver</li> <li>• zwei DBMS</li> </ul>
Groß	Mehr als 500	<ul style="list-style-type: none"> <li>• mehrere Windows/Samba Domäne</li> <li>• ein Office Produkt je User</li> <li>• zehn Datei- /Druckserver</li> <li>• drei DBMS</li> </ul>

Diese Werte wurden dem Migrationsleitfaden der KBSt (BMI) Seite 315 entnommen und leicht abgeändert, da diese Thesis andere Organisationsgrößen definiert hat.

Folgende Produkte werden bei den Beispielen verwendet:

Ebene	Vertretendes heutiges Szenario	Empfohlenes/künftiges Szenario Migrationsformen / betroffene Produkte					
		Vollständige Migration		Fortführende Migration	Teil-Migration		
		groß	mittel-klein	klein-mittel-groß	Punktuelle Migration	Breite Migration	
Client	Infrastruktur	MS-Windows NT WS	Linux, Linux-Distribution [Debian, Suse, RedHat,...]	Linux, Linux-Distribution [Debian, Suse, RedHat,...]	MS Windows 2000		
	Desktop	MS-Windows NT WS	KDE, Gnome	KDE, Gnome	MS Windows 2000 WS		
	Office	MS-Office 97	Open Office, Staroffice	Open Office, Staroffice	MS Office 2000	Open Office, Staroffice	
Server	Infrastruktur - Server	MS Windows NT 4.0 Server	Linux Server	Linux Server	MS Windows 2000 Server		Linux Server
	Infrastruktur - Web	MS Windows NT 4.0 Server	Webeerver Apache	Webeerver Apache	MS Windows 2000 Server		Webeerver Apache
	Infrastruktur - Dateiablage	MS Windows NT 4.0 Server	XFS	XFS	MS Windows 2000 Server		Samba
	Infrastruktur - Druckdienst	MS Windows NT 4.0 Server	CUPS	CUPS	MS Windows 2000 Server		CUPS, Samba
	Infrastruktur - Netzwerk	MS Windows NT 4.0 Server	BIND, ...	BIND, ...	MS Windows 2000 Server		BIND, ...
	Datenbank Mgmt. Syst.	MS SQL 7.0	SAP DB, Oracle, DB2	SAP DB, MY SQL, postgresQL	MS SQL 2000 Server		SAP DB, MY SQL, postgresQL
	Messaging/ Groupware	MS Exchange 5.5	SamsungContact	SamsungContact, Kroupware	MS Exchange 2000	SamsungContact, Exchange4linux	Samsung, Exchange4linux
	Verzeichnisdienst	--	Sun ONE	OpenLDAP	--		Sun One, OpenLDAP

Hinweis: Produkte in Fettschrift sind kostenlos zu erwerben.

Diese Werte wurden dem Migrationsleitfaden der KBSt (BMI) Seite 342 entnommen.

Die vergleichende Betrachtung der Migrationsalternativen wird ausgehend von der aktuellen Umgebung gemacht. Das Ausgangsszenario ist dabei eine Microsoft-basierte Plattform. Einzig die fortführende Migration beinhaltet keine rechenbaren Ersparnisse. Sämtliche anderen Varianten enthalten entsprechende nicht verausgabte Aufwendungen als Ersparnisse.

Die Hardwarekosten werden vernachlässigt, da davon ausgegangen wird, dass eine gleichwertige Hardwareausstattung sowohl für Windows als auch für Linux verwendet wird.

Als Lizenzkosten werden folgende Werte herangezogen:

- Windows 2003 Server: 192,- Euro pro Arbeitsplatz
- Office XP: 589,- Euro pro Arbeitsplatz
- SQL Server: 285,- Euro pro Arbeitsplatz
- Exchange: 57,- Euro pro Arbeitsplatz

Lizenzsumme: 1123,- / 2 = ca. 560,- Euro pro Arbeitsplatz

Die Lizenzgebühren wurden einschlägigen Zeitschriften / Online Shops entnommen und durch 2 dividiert!

Für den OSS Server wird SUSE Linux Enterprise Server verwendet. Die einmaligen Kosten belaufen sich auf: 850,- Euro für alle User!

Für das Mietmodell werden Kosten von 79 € je Benutzer für kleine Unternehmen und 149 € je Benutzer mittlerer und großer Unternehmen, je Monat genommen. Dieser Wert stammt aus der Bachelorthesis von Janko (2002), Seite 78 und beruht auf einen Wert der Firma Einsteinet GmbH (Deutschland). Für die hohen Benutzerzahlen wurde ein Abschlag angenommen, der im angenommenen Preis bereits beinhaltet ist.

## Vollständige Migration

### Projekt: vollständige Migration von Windows NT auf Linux

Unternehmenstyp:	Klein	Mittel	Groß
Anzahl User:	100	250	500
Prozentsatz für Abzinsung:	5%		
Betrachtungszeitraum	jährlich		

#### Vollständige Migration für kleine Unternehmen

Saldo Kosten-Nutzen	Kostenverteilung	
Hardware Aufwand	0	
Software Aufwand	€ 810	1,52%
Personal Aufwand	€ 52.534	98,48%
Gesamtkosten	€ 53.344	
Gesamtkosten je User	€ 533	

#### Vollständige Migration für mittlere Unternehmen

Saldo Kosten-Nutzen	Kostenverteilung	
Hardware Aufwand	0	
Software Aufwand	€ 810	0,89%
Personal Aufwand	€ 90.516	99,11%
Gesamtkosten	€ 91.326	
Gesamtkosten je User	€ 365	

#### Vollständige Migration für große Unternehmen

Saldo Kosten-Nutzen	Kostenverteilung	
Hardware Aufwand	0	
Software Aufwand	€ 810	0,50%
Personal Aufwand	€ 160.460	99,50%
Gesamtkosten	€ 161.270	
Gesamtkosten je User	€ 323	

## Fortführende Migration

### Projekt: fortführende Migration von Windows

### NT auf Windows 2003/XP

Unternehmenstyp:	Klein	Mittel	Groß
Anzahl User:	100	250	500
Prozentsatz für Abzinsung:	5%		
Betrachtungszeitraum	jährlich		

#### Vollständige Migration für kleine Unternehmen

Saldo Kosten-Nutzen		Kostenverteilung
Hardware Aufwand	0	
Software Aufwand	€ 53.334	61,84%
Personal Aufwand	€ 32.915	38,16%
Gesamtkosten	€ 86.249	
Gesamtkosten je User	€ 862	

#### Vollständige Migration für mittlere Unternehmen

Saldo Kosten-Nutzen		Kostenverteilung
Hardware Aufwand	0	
Software Aufwand	€ 133.336	64,29%
Personal Aufwand	€ 74.059	35,71%
Gesamtkosten	€ 207.395	
Gesamtkosten je User	€ 830	

#### Vollständige Migration für große Unternehmen

Saldo Kosten-Nutzen		Kostenverteilung
Hardware Aufwand	0	
Software Aufwand	€ 266.672	64,94%
Personal Aufwand	€ 144.003	35,06%
Gesamtkosten	€ 410.675	
Gesamtkosten je User	€ 821	

## Teilmigration

### Projekt: serverseitige Teilmigration von Windows NT auf Linux

Unternehmenstyp:	Klein	Mittel	Groß
Anzahl User:	100	250	500
Prozentsatz für Abzinsung:	5%		
Betrachtungszeitraum	jährlich		

#### Vollständige Migration für kleine Unternehmen

Saldo Kosten-Nutzen		Kostenverteilung
Hardware Aufwand	0	
Software Aufwand	€ 810	2,73%
Personal Aufwand	€ 28.801	97,27%
Gesamtkosten	€ 29.610	
Gesamtkosten je User	€ 296	

#### Vollständige Migration für mittlere Unternehmen

Saldo Kosten-Nutzen		Kostenverteilung
Hardware Aufwand	0	
Software Aufwand	€ 810	1,20%
Personal Aufwand	€ 66.439	98,80%
Gesamtkosten	€ 67.249	
Gesamtkosten je User	€ 269	

#### Vollständige Migration für große Unternehmen

Saldo Kosten-Nutzen		Kostenverteilung
Hardware Aufwand	0	
Software Aufwand	€ 810	0,65%
Personal Aufwand	€ 123.431	99,35%
Gesamtkosten	€ 124.241	
Gesamtkosten je User	€ 248	

## Projekt: punktuelle Teilmigration von Windows NT auf Linux

Unternehmenstyp:	Klein	Mittel	Groß
Anzahl User:	100	250	500
Prozentsatz für Abzinsung:	5%		
Betrachtungszeitraum	jährlich		

### Vollständige Migration für kleine Unternehmen

Saldo Kosten-Nutzen		Kostenverteilung
Hardware Aufwand	0	
Software Aufwand	€ 810	10,01%
Personal Aufwand	€ 7.276	89,99%
Gesamtkosten	€ 8.086	
Gesamtkosten je User	€ 81	

### Vollständige Migration für mittlere Unternehmen

Saldo Kosten-Nutzen		Kostenverteilung
Hardware Aufwand	0	
Software Aufwand	€ 810	7,20%
Personal Aufwand	€ 10.438	92,80%
Gesamtkosten	€ 11.248	
Gesamtkosten je User	€ 45	

### Vollständige Migration für große Unternehmen

Saldo Kosten-Nutzen		Kostenverteilung
Hardware Aufwand	0	
Software Aufwand	€ 810	5,27%
Personal Aufwand	€ 14.553	94,73%
Gesamtkosten	€ 15.362	
Gesamtkosten je User	€ 31	