

Peter Buxmann
Heiner Diefenbach
Thomas Hess

Die Softwareindustrie

Ökonomische Prinzipien,
Strategien, Perspektiven

3. Auflage

Die Softwareindustrie

Peter Buxmann • Heiner Diefenbach
Thomas Hess

Die Softwareindustrie

Ökonomische Prinzipien, Strategien,
Perspektiven

3., vollständig überarbeitete und erweiterte Auflage

Peter Buxmann
Wirtschaftsinformatik | Software Business &
Information Management
Technische Universität Darmstadt
Darmstadt
Deutschland

Thomas Hess
Wirtschaftsinformatik und Neue Medien
LMU München
München
Deutschland

Heiner Diefenbach
TDS AG
Neckarsulm
Deutschland

ISBN 978-3-662-45588-3
DOI 10.1007/978-3-662-45589-0

ISBN 978-3-662-45589-0 (eBook)

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

Springer Gabler

© Springer-Verlag Berlin Heidelberg 2008, 2011, 2015

Das Werk einschließlich aller seiner Teile ist urheberrechtlich geschützt. Jede Verwertung, die nicht ausdrücklich vom Urheberrechtsgesetz zugelassen ist, bedarf der vorherigen Zustimmung des Verlags. Das gilt insbesondere für Vervielfältigungen, Bearbeitungen, Übersetzungen, Mikroverfilmungen und die Einspeicherung und Verarbeitung in elektronischen Systemen.

Die Wiedergabe von Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. in diesem Werk berechtigt auch ohne besondere Kennzeichnung nicht zu der Annahme, dass solche Namen im Sinne der Warenzeichen- und Markenschutz-Gesetzgebung als frei zu betrachten wären und daher von jedermann benutzt werden dürften. Der Verlag, die Autoren und die Herausgeber gehen davon aus, dass die Angaben und Informationen in diesem Werk zum Zeitpunkt der Veröffentlichung vollständig und korrekt sind. Weder der Verlag noch die Autoren oder die Herausgeber übernehmen, ausdrücklich oder implizit, Gewähr für den Inhalt des Werkes, etwaige Fehler oder Äußerungen.

Gedruckt auf säurefreiem und chlorfrei gebleichtem Papier

Springer Fachmedien Wiesbaden GmbH ist Teil der Fachverlagsgruppe Springer Science+Business Media (www.springer.com)

Vorwort zur ersten Auflage

Kaum eine andere Branche hat die Gesellschaft und die Unternehmenswelt so nachhaltig verändert wie die Softwareindustrie. So ist die Softwareunterstützung von inner- sowie zwischenbetrieblichen Geschäftsprozessen heute genauso eine Selbstverständlichkeit wie das „Googeln“ nach Informationen oder die Nutzung von Navigationsgeräten. Es ist daher auch wenig überraschend, dass die Softwareindustrie gemäß IDC bei einem weltweiten jährlichen Umsatz von 230 Mrd. US-Dollar (ohne komplementäre Dienstleistungen) und mit einem Zuwachs von jährlich sieben Prozent in Europa sowie bis zu neun Prozent in Asien zu den bedeutendsten Wachstumsmärkten zählt. Dabei gilt, dass die Softwarebranche so international wie kaum eine andere ist. Dementsprechend konkurrieren Softwareunternehmen weltweit um Kunden und zunehmend auch um Mitarbeiter. Aber nicht nur aufgrund der internationalen Softwaremärkte, sondern auch wegen der besonderen Eigenschaften des Gutes Software gelten für Softwareanbieter eigene Spielregeln. Hier von handelt dieses Buch.

Es wäre aufgrund der Besonderheiten und auch der Attraktivität der Branche zu erwarten, dass es – ähnlich wie etwa für den Industrie- und Bankensektor, für Versicherungen oder für die Medien- und Energiewirtschaft – eine spezielle Betriebswirtschaftslehre oder Lehrstühle für die Softwareindustrie aus der Wirtschaftsinformatik heraus gibt. Dies ist bisher jedoch überraschenderweise nicht der Fall. Diese Lücke war für uns ein Anlass, die Software Economics Group Darmstadt-München (www.software-economics.org) zu gründen sowie das vorliegende Buch zu schreiben.

Im ersten Kapitel werden wir nach einer kurzen Beschreibung der Anfänge der Softwareindustrie auf die verschiedenen Player in dieser Branche eingehen. Einen Schwerpunkt bildet die Untersuchung der unterschiedlichen Erlösquellen, also das Lizenz- und Servicegeschäft. Die ökonomischen Prinzipien der Softwareindustrie werden im zweiten Kapitel thematisiert. Hierzu gehören die spezifischen Eigenschaften des Gutes Software sowie die besonderen Merkmale von Softwaremärkten. Darauf aufbauend werden im dritten Kapitel Strategieempfehlungen für die Softwareindustrie abgeleitet. Dabei betrachten wir Kooperations-, Preis-, Vertriebs- und Industrialisierungsstrategien. In diesem Zusammenhang werden wir auch neue Technologien, wie das SOA-Konzept und das Model Driven Engineering, untersuchen. Dabei liegt der Schwerpunkt allerdings eher auf der Analyse der Auswirkungen auf die Softwareindustrie als auf der Darstellung technischer

Implementierungsdetails. Aufgrund des hohen Internationalisierungsgrads der Branche gehen wir im vierten Kapitel auf die Globalisierung der Softwareentwicklung ein. Dabei werden wir uns mit Offshore- sowie Open-Source-Projekten und ihren Auswirkungen auf die Softwareindustrie beschäftigen.

Bei der Erstellung des Buches konnten wir auf die Arbeiten der bereits oben angesprochenen Software Economics Group Darmstadt-München zurückgreifen. Im Rahmen dieses Projekts ist auch das empirische Material entstanden, das eine wichtige Grundlage verschiedener Abschnitte ist. So haben wir eine Vielzahl von Expertengesprächen geführt. Wir möchten uns sehr herzlich bei den folgenden Personen aus der Softwareindustrie dafür bedanken, dass sie ihr Wissen und ihre Erfahrungen mit uns geteilt bzw. uns auf andere Weise unterstützt haben: Dieter Berz, Rüdiger Bolt, Christian Boos, Debjit Chaudhuri, Akhil Eswaran, Alexander Gassmann, Martin Haas, Jürgen Henn, Martin Helling, Lutz Heuser, Franz Hollich, Kim Langer, Horst Kinzinger, Thomas Lünendonk, Knut Manske, Thomas Meyer, Karsten Öhler, Jens Pfennig, Jürgen Powik, Nikolai Puntikov, Ganesh Ramaoorthy, Andreas Reinicke, Stefan Ried, Morten Rolfes, Thomas Roth, Karl-Heinz Streibich, Christopher Sürrie, Ferenc Szilágyi, Justin Vaughan-Brown, Joachim Voegele und Martin Winkler.

Unseren Mitarbeiterinnen und Mitarbeitern Björn Brandt, Julia Gebele, Markus Hahn, Patrick Johnscher, Sven Schade, Jochen Strube, Thomas Widjaja und Christian Wolf danken wir für ihre fachliche Unterstützung bei der Erstellung dieses Buches.

Bei Anette von Ahsen, Melanie Hüttenberger und Gerrit Pohl bedanken wir uns sehr herzlich für die vielen und umfangreichen „Bug Reports“. Ebenso dankbar sind wir Niels Thomas vom Springer-Verlag für die sehr angenehme Zusammenarbeit und die vielen guten Verbesserungsvorschläge.

Wir freuen uns auf Ihre Kommentare und Anmerkungen und wünschen Ihnen viele Anregungen und auch viel Spaß beim Lesen des Buchs.

im September 2007

Peter Buxmann
Heiner Diefenbach
Thomas Hess

Vorwort zur dritten Auflage

Die Softwareindustrie ist eine sich ständig verändernde Branche. Mit dieser dritten Auflage wollen wir das bewährte Konzept der beiden vorigen Auflagen fortführen und ein aktuelles Bild der Softwareindustrie zeichnen. Seit der zweiten Auflage dieses Buches in 2011 haben brancheninterne Veränderungen und der wissenschaftliche Dialog neue Erkenntnisse hervorgebracht. Um diese Entwicklungen zu berücksichtigen, haben wir insbesondere die Teile zu den Themenbereichen Cloud Computing und Geschäftsmodelle in der Softwareindustrie deutlich ausgeweitet. Darüber hinaus wurden neue Teilabschnitte verfasst, u. a. zur Abhängigkeit in IT-Outsourcing-Beziehungen, zum Fachkräftemangel in der Software- und IT-Industrie, zur Bewertung des M&A-Erfolges in der Softwareindustrie sowie zur Erschließung internationaler Märkte.

Für die fachliche Unterstützung zur Erstellung dieser dritten Auflage bedanken wir uns sehr herzlich bei unseren Mitarbeiterinnen und Mitarbeitern der Software Economics Group Darmstadt-München (www.software-economics.org). So sind viele Neuerungen durch Inhaltsbeiträge von Dr. Tobias Ackermann, Dr. Stefan Harnisch, Dr. Jasmin Kaiser, Natalie Kaltenecker, André Loske, Anton Pussep und Dr. Markus Schief entstanden.

Auch bedanken wir uns erneut sehr herzlich bei Anette von Ahsen für ihre hervorragende Unterstützung durch umfangreiche „Bug Reports“ und die dazu passenden „Patches“. Ebenso dankbar sind wir Miriam Bär und Rabea Sonnenschein für ihre außerordentliche Unterstützung bei der Manuskripterstellung. Herrn Dr. Niels Peter Thomas vom Springer-Verlag möchten wir ebenfalls danken für die erneut sehr angenehme Zusammenarbeit.

Wir freuen uns auf Ihre Kommentare und Anmerkungen und wünschen Ihnen viele Anregungen und auch viel Spaß beim Lesen dieses Buchs.

im September 2014

Peter Buxmann
Heiner Diefenbach
Thomas Hess

Inhaltsverzeichnis

Teil I Allgemeine Grundlagen

1 Die Spielregeln in der Softwareindustrie	3
1.1 Software und Softwaremärkte: Ausgewählte Besonderheiten der Softwareindustrie im Überblick	3
1.2 Die Anfänge der Softwareindustrie.	4
1.3 Typen von Softwareanbietern und Auswahlentscheidungen der Anwender	5
1.3.1 Softwareanbieter im engeren und weiteren Sinne	5
1.3.2 Die Auswahl von Software	10
1.4 Dienstleistungserlöse in der Softwareindustrie.	16
Literatur	17
2 Ökonomische Prinzipien der Softwareindustrie	19
2.1 Eigenschaften digitaler Güter	19
2.2 Netzeffekte auf Softwaremärkten: The Winner Takes it All.	20
2.2.1 Netzeffekte: Grundlagen und Definitionen	21
2.2.2 Auswirkungen von Netzeffekten auf Softwaremärkte	24
2.2.3 Struktur von Softwaremärkten	27
2.2.4 Netzeffekte als Wettbewerbsfaktor	29
2.2.5 Ein Anwendungsbeispiel: Zweiseitige Netzeffekte und Plattformstrategien in der digitalen Spieleindustrie	30
2.2.6 Grenzen der Netzeffekttheorie	34
2.3 Das Standardisierungsproblem	35
2.3.1 Ansatz und Hintergründe	35
2.3.2 Das zentrale Standardisierungsproblem als Optimierungsproblem	38
2.3.3 Das dezentrale Standardisierungsproblem – Eine spieltheoretische Darstellung	40
2.3.4 Das Standardisierungsproblem – Lessons learned	42
2.4 Transaktionskostentheorie: Auf der Suche nach den Grenzen eines Softwareunternehmens	44
2.4.1 Ansatzpunkt und Elemente der Transaktionskostentheorie	45

2.4.2	Arbeitsteilung zwischen Unternehmen aus Sicht der Transaktionskostentheorie.	47
2.4.3	Strukturelle Veränderungen der Transaktionskosten: The Move to the Middle.	48
2.4.4	Ausblick: Intermediäre und Transaktionskosten.	50
2.5	Softwareentwicklung als Agency-Problem: Anreizkompatible Entlohnung und effiziente Kontrolle.	51
2.5.1	Principal-Agent-Beziehungen: Definitionen und Grundlagen	51
2.5.2	Anreizkompatible Vergütungsschemata	53
2.5.3	Kontrollsysteme	56
	Literatur	58
3	Strategien für Softwareanbieter	61
3.1	Kooperations- und Übernahmestrategien	61
3.1.1	Kooperationen in der Softwareindustrie	62
3.1.2	Mergers & Acquisitions in der Softwareindustrie.	72
3.2	Vertriebsstrategien.	85
3.2.1	Gestaltung des Vertriebssystems: Organisation und Vertriebswege in der Softwareindustrie	85
3.2.2	Gestaltung der Beziehungen zu Vertriebspartnern und Key Accounts	90
3.2.3	Kennzahlensysteme als Instrument für das Vertriebscontrolling in der Softwareindustrie	92
3.2.4	Gestaltung der Verkaufsaktivitäten.	96
3.2.5	Erschließung internationaler Märkte.	102
3.3	Preisstrategien.	107
3.3.1	Grundüberlegungen.	107
3.3.2	Parameter der Preisgestaltung für Softwareprodukte	108
3.3.3	Ansätze zur Preissetzung für Individualsoftwareanbieter.	122
3.4	Entwicklungsstrategien.	125
3.4.1	Strukturierung der Softwareentwicklung	125
3.4.2	Personalführung in der Softwareentwicklung.	129
3.4.3	Add-on: Fachkräftemangel in der Software- und IT-Industrie	132
	Literatur	135
4	Geschäftsmodelle in der Softwareindustrie.	141
4.1	Die Wertschöpfungskette in der Softwareindustrie	142
4.1.1	Kernaktivitäten der Software-Wertschöpfungskette	142
4.1.2	Wertschöpfungsketten in der Softwareindustrie – 3 Fallstudien.	144
4.2	Geschäftsmodelle in der Softwareindustrie – ein Framework	152
4.3	Software Business Model Tool.	158
	Literatur	163

Teil II Spezielle Themen

5 Outsourcing und Offshoring der Softwareentwicklung	169
5.1 Überblick	169
5.2 Formen des Outsourcings und Offshorings	170
5.3 Motive für Outsourcing und Offshoring	174
5.4 Abhängigkeit in IT-Outsourcing-Beziehungen	178
5.4.1 Grundlagen der Analyse	178
5.4.2 Relevanz	181
5.4.3 Substituierbarkeit	182
5.4.4 Spillover	184
5.5 Standortwahl von Softwareanbietern	185
5.6 Outsourcing durch Softwareanwender	188
5.6.1 Outsourcing der Neuentwicklung von Individualsoftware	188
5.6.2 Outsourcing der Anpassung von Standardsoftware	191
5.6.3 Outsourcing der Weiterentwicklung und Wartung von Anwendungssoftware	194
5.6.4 Zufriedenheit der Anwender mit Onshore-, Nearshore- und Farshoreanbietern	198
5.7 Nearshoring versus Farshoring: Die Entfernung zum Kunden als Erfolgsfaktor?	199
5.7.1 Sprachliche und kulturelle Barrieren in Offshore-Projekten	199
5.7.2 Die Bedeutung persönlicher Treffen für den Projekterfolg	202
5.7.3 Herausforderungen und Chancen der Zeitverschiebung	203
Literatur	205
6 Plattformkonzepte	207
6.1 Überblick	207
6.2 Produktplattformen in der Softwareindustrie	207
6.2.1 Kostenstruktur plattformbasierter Softwareentwicklung	207
6.2.2 Add-on: Industrialisierung als Managementkonzept für die Softwareindustrie	210
6.3 Branchenplattformen in der Softwareindustrie	213
6.3.1 Offenheit einer Branchenplattform	213
6.3.2 Das Management der Komplementäre	216
Literatur	219
7 Cloud Computing	221
7.1 Überblick	221
7.2 Der Cloud-Markt	223
7.3 Die Sicherheit in der Cloud – Empirische Ergebnisse aus Anwender- und Anbieterperspektive	225

7.3.1	Anwendersicht	226
7.3.2	Anbietersicht	229
7.4	Software as a Service: die Anwendungsebene des Cloud Computing	231
7.4.1	SaaS aus Anwendersicht – Chancen und Risiken	233
7.4.2	SaaS aus Anbietersicht – Chancen und Risiken	240
7.4.3	Empirische Untersuchung der Preisstrategien und Geschäftsmodelle für SaaS-Anbieter	247
7.4.4	Fallstudie zum Vergleich nutzungsabhängiger und nutzungsunabhängiger Preismodelle	252
	Literatur	254
8	Open Source Software	257
8.1	Überblick	257
8.2	Charakteristika von Open Source Software	258
8.3	Open-Source-Projekte: Prinzipien und Motivation der Softwareentwickler	262
8.3.1	Ablauf und Organisation von Open-Source-Projekten	262
8.3.2	Zur Motivation der Beitragenden	264
8.4	Open Source Software aus Sicht des Anwenders	266
8.5	Engagement kommerzieller Softwareanbieter	267
8.6	Quelloffene ERP-Systeme	269
	Literatur	277
	Weiterführende Literatur	279
	Sachverzeichnis	291

Abkürzungsverzeichnis

AGB	Allgemeine Geschäftsbedingungen
ASD	Adaptive Software Development
ASP	Application Service Providing
BOT	Build Operate Transfer
BSA	Business Software Alliance
BSD	Berkeley Software Distribution
CAR	Cumulative Abnormal Return
CAAR	Cumulative Average Abnormal Return
CCC	Content Communication and Collaboration
CEO	Chief Executive Officer
CIO	Chief Information Officer
CMMI	Capability Maturity Model Integration
CRM	Customer Relationship Management
CTO	Chief Technology Officer
CVS	Concurrent Versions System
DACH	Deutschland, Österreich, Schweiz
EDI	Electronic Document Interchange
EDV	Elektronische Datenverarbeitung
ERP	Enterprise Resource Planning
GE	Geldeinheiten
GNU	GNU is not Unix
GPL	General Public License
HTTP	Hypertext Transfer Protocol
IE	Internet Explorer
IP	Intellectual Property
ISO	International Organization for Standardization
KMU	Kleine und mittelständische Unternehmen
KPI	Key Performance Indicator
LGPL	Library/Lesser General Public License
LoC	Lines of Code
M&A	Mergers & Acquisitions

MIT	Massachusetts Institute of Technology
MPL	Mozilla Public License
OEM	Original Equipment Manufacturer
OSI	Open Source Initiative
OSS	Open Source Software
PDF	Portable Document Format
PMC	Point of Marginal Cheapness
PME	Point of Marginal Expensiveness
PPS	Produktionsplanungs- und -steuerungssystem
PSM	Price Sensitivity Meter
ROI	Return on Investments
SaaS	Software as a Service
SIIA	Software & Information Industry Association
SCM	Supply Chain Management
SOA	Serviceorientierte Architektur
SQL	Structured Query Language
SW	Software
VB	Vertriebsbeauftragte(r)
VHS	Video Home System
XP	eXtreme Programming
XML	Extensible Markup Language
ZB	Zahlungsbereitschaft

Teil I
Allgemeine Grundlagen

1.1 Software und Softwaremärkte: Ausgewählte Besonderheiten der Softwareindustrie im Überblick

Die Softwareindustrie unterscheidet sich grundsätzlich von anderen Branchen. Dies ist zum einen auf die spezifischen Eigenschaften des Produkts Software, zum anderen auf die Struktur von Softwaremärkten zurückzuführen.

Eine Besonderheit von Softwareprodukten besteht darin, dass sie sich, wie jedes andere digitale Gut auch, zu geringen Kosten reproduzieren lassen. Die variablen Kosten gehen also gegen Null. Diese Kostenstruktur führt etwa dazu, dass das Lizenzgeschäft eines Softwareanbieters – zumindest auf den ersten Blick – in der Regel profitabler ist als das Servicegeschäft, wie wir noch ausführlich diskutieren werden. Darüber hinaus kann Software beliebig häufig und ohne Qualitätsverluste kopiert werden. Ist eine Kopie im Internet erst einmal im Umlauf, lassen sich Urheber- bzw. Verfügungsrechte faktisch nicht mehr durchsetzen. Dies gilt insbesondere für wenig erklärungsbedürftige Produkte auf Business-to-Consumer-Märkten. Es ist zudem relativ einfach, von einem einmal erstellten Softwareprodukt verschiedene Versionen oder Bündel zu erstellen und diese zu unterschiedlichen Preisen an verschiedene Kundengruppen zu verkaufen.

Auch für Softwaremärkte gelten einige Besonderheiten. Wie kaum eine andere Branche ist die Softwareindustrie durch eine starke Internationalisierung gekennzeichnet. Software lässt sich global verteilt entwickeln und in Sekundenschnelle zu vernachlässigbaren Kosten über das Internet vertreiben. Daraus resultiert auch ein weltweiter Wettbewerb zwischen den Softwareanbietern. Im Vergleich zu anderen Branchen spielt hierbei der Heimvorteil auf den nationalen Märkten der Anbieter in vielen Segmenten nur noch eine untergeordnete Rolle. So erzielen deutsche Softwareanbieter im Durchschnitt mehr als die Hälfte ihrer Umsätze im Ausland. Der Exportanteil am Umsatz der beiden größten deutschen Softwareanbieter SAP und Software AG betrug in 2012 circa 85 bzw. 80%

(Lünendonk 2013). Zudem führt der Netzeffektcharakter von Software dazu, dass es sich bei Softwaremärkten häufig um so genannte „Winner-takes-it-all“-Märkte handelt. Vor diesem Hintergrund lässt sich beispielsweise auch die Vielzahl von Unternehmensübernahmen erklären.

Diese und andere spezielle ökonomische Prinzipien und Spielregeln werden wir aufgreifen und ausführlich untersuchen. Sie bilden letztlich die Grundlage für die Formulierung von Strategien und Geschäftsmodellen für die Softwareindustrie.

Doch zunächst wollen wir uns in aller Kürze mit der historischen Entwicklung der Softwareindustrie beschäftigen.

1.2 Die Anfänge der Softwareindustrie

Die Softwareindustrie ist eine relativ junge Branche. Die Anfänge gehen auf die frühen fünfziger Jahre zurück, als es noch üblich war, Software und Hardware zu bündeln und gemeinsam zu verkaufen. Die Software war damals also integrierter Bestandteil der Hardware und es wurde noch ausschließlich von Programmcode gesprochen – der Begriff Software wurde erstmals im Jahr 1959 verwendet (Campbell-Kelly 1995, S. 80). In dieser Zeit entstanden in den USA die ersten kleineren Firmen, die Programmcode bzw. Software im Rahmen von individuellen Auftragsprojekten entwickelten (Hoch et al. 2000, S. 27 f.).

Eine Aufwertung der Software erfolgte 1969, als das amerikanische Justizministerium von IBM den getrennten Ausweis von Hardware und Software auf Rechnungen verlangte. In den siebziger Jahren entstand daraufhin eine Reihe von Unternehmen, die sich ausschließlich mit der Entwicklung von Software beschäftigten. Hierbei ist natürlich in erster Linie Microsoft zu nennen: Das von Bill Gates zusammen mit Paul Allen gegründete Unternehmen begann anfangs Programmiersprachen – zunächst BASIC, dann weitere, wie FORTRAN und COBOL – für verschiedene Prozessoren und Rechner zu entwickeln. Erst später entstand im Rahmen einer Zusammenarbeit mit IBM MS-DOS, das zum Standard für Betriebssysteme wurde und wesentlich zur Verbreitung des Personal Computers beitrug (Ichbiah 1993, S. 91–116). Schließlich entschied Microsoft, auch Anwendungen anzubieten und damit in Konkurrenz z. B. zu Lotus zu treten. Bereits 1983 kündigte Bill Gates in der Zeitschrift *Business Week* an, dass es das Ziel von Microsoft sei, alle PC-Software zukünftig aus einer Hand anzubieten (Ichbiah 1993, S. 141).

Eine parallele Erfolgsgeschichte begann etwa zeitgleich mit den Anfängen von Microsoft im badischen Walldorf, als Dietmar Hopp, Hans-Werner Hector, Hasso Plattner, Klaus Tschira und Claus Wellenreuther ein Unternehmen gründeten, das sich auf die Entwicklung von Software für betriebswirtschaftliche Funktionen und Prozesse spezialisierte – die SAP AG war geboren. Zunächst stellte die Firma Software für Großrechner her, später wurden die Anwendungen für Client-Server-Umgebungen angeboten. Heute ist die SAP das größte europäische Softwarehaus und weltweiter Marktführer im Bereich von Enterprise Resource Planning Software (ERP-Software).

Bereits an diesen beiden Beispielen lässt sich eine Besonderheit der Softwareindustrie erkennen: Am Markt setzt sich häufig nur eine Technologie bzw. ein Anbieter durch. So

verdrängte MS-DOS das Betriebssystem CPM ebenso, wie sich das Tabellenkalkulationsprogramm Excel gegen Konkurrenzprodukte wie Lotus 1-2-3 durchsetzte. Mittlerweile ist Microsoft weltweit führender Anbieter von Office-Anwendungen, Browsern und Betriebssystemen. Ebenso findet auch auf dem Markt für betriebswirtschaftliche Software aktuell eine Konsolidierung statt (siehe hierzu Abschn. 3.1.2). Wir werden auf die Besonderheiten dieser Märkte zurückkommen, wollen uns aber zunächst mit den Akteuren der Softwareindustrie beschäftigen.

1.3 Typen von Softwareanbietern und Auswahlentscheidungen der Anwender

1.3.1 Softwareanbieter im engeren und weiteren Sinne

Betrachten wir im Folgenden die verschiedenen Typen von Softwareanbietern. Hierbei soll zwischen Softwareanbietern im engeren und im weiteren Sinne unterschieden werden.

Die Aufgabe eines Softwareanbieters im engeren Sinne ist die Entwicklung von Software. Dies gilt unabhängig davon, um welche Art von Software es sich hierbei handelt. Software lässt sich nach verschiedenen Merkmalen differenzieren. Ein häufig verwendetes Kriterium ist die Nähe zur Hardware (Mertens et al. 2005). Demnach kann Software in Systemsoftware (z. B. Betriebssysteme), systemnahe Software (z. B. Datenbanksysteme oder Middleware) und Anwendungssoftware (z. B. für Textverarbeitung oder für das Rechnungswesen) unterschieden werden. Ebenfalls lässt sich zwischen Software für kommerzielle und private Anwender unterscheiden. Ein drittes Kriterium zur Klassifikation von Software, das für unsere Betrachtung wesentlich wichtiger ist, ist der Standardisierungsgrad. Als Extremformen sind hier Individualsoftware und Standardsoftware zu nennen.

Individualsoftware wird auf der Basis von spezifischen Anforderungen maßgeschneidert entwickelt, und zwar entweder von den anwendenden Unternehmen selbst – in der Regel in der IT-, in manchen Fällen aber auch in den jeweiligen Fachabteilungen – oder von einem externen Softwarehaus. Die Ausgestaltung der Verträge im Falle der Inanspruchnahme eines solchen Drittanbieters ist jedoch sehr unterschiedlich. Nach nahezu einhelliger Auffassung handelt es sich bei Projekten zur Individualsoftwareherstellung um Werkverträge (Marly 2014, S. 236 f.). Auf eine ökonomische Untersuchung von Dienst- und Werkverträgen gehen wir in Abschn. 2.5 ein. Insbesondere in Indien boomt die Branche im Bereich der Individualsoftwareentwicklung. So haben sich einige erfolgreiche Softwareanbieter mit enormen Wachstumsraten etablieren können, wie etwa Tata Consultancy Services, Wipro Technologies, Infosys Technologies sowie Cognizant Technology Solutions. Daneben existieren aber auch viele kleinere oder mittelständische Firmen, die in Hochlohnländern häufig noch national bzw. regional tätig sind. Demgegenüber sind die Individualsoftwareanbieter mit Sitz in Niedriglohnländern in den meisten Fällen global aufgestellt und stehen miteinander im Wettbewerb. Wir werden auf diese Nearshore- und

Farshore-Anbieter und die sich daraus ergebenden Auswirkungen auf die Softwareindustrie in Abschn. 5.7 näher eingehen.

Standardsoftware wird in der Regel für den Massenmarkt entwickelt. Dabei gehen die Anbieter von weitestgehend standardisierten Bedürfnissen der potenziellen Nutzer aus. Im Folgenden wollen wir kurz die Entstehung und Erfolgsfaktoren von Standardsoftware am Beispiel der SAP AG betrachten.

Das System R von SAP

Die SAP AG ist laut eigenen Angaben führender Anbieter von Unternehmenssoftware und drittgrößter unabhängiger Softwarelieferant der Welt. Weltweit beschäftigt die SAP 66.000 Mitarbeiter und hat Standorte in mehr als 50 Ländern. Die SAP-Lösungen sind bei 251.000 Kunden im Einsatz.

Kernaspekte

Der Erfolg von SAP lässt sich auf drei Kernaspekte zurückführen: die Idee der Entwicklung von Standardsoftware, das Anbieten integrierter Lösungen und Echtzeitverarbeitung. 1981 implementierte das Unternehmen diese Visionen zum ersten Mal mit SAP R/2 und legte seinen Schwerpunkt damit auf die Entwicklung betriebswirtschaftlicher Systeme. 1991 präsentierte SAP das System R/3, das ursprünglich nicht als Ablösung für R/2 gedacht war, sondern dieses als Mittelstandslösung ergänzen sollte. Der generelle Wechsel von Großrechnerlösungen zum Client-Server-Prinzip, der zu dieser Zeit im Gange war, machte R/3 aber gerade für größere Unternehmen interessant und verhalf dem System damit zum Erfolg. Mittlerweile hat SAP seine Produktpalette weiterentwickelt, worauf wir aber an anderer Stelle eingehen.

Standardsoftware

Wie bereits erwähnt, wurde Software anfangs meistens im Rahmen von Individualprojekten entwickelt. Dabei wurden die Software und gegebenenfalls auch die Hardware exakt auf die Anforderungen des Auftraggebers zugeschnitten. Demgegenüber plante SAP von Anfang an die Entwicklung eines standardisierten Systems, das sich mehrfach einsetzen lässt. SAP war eines der ersten Unternehmen, die dieses Konzept konsequent verfolgten.

Integrierte Lösung

Grundlage der Anwendung eines SAP-Systems ist eine integrierte Datenbank, auf die alle Anwendungen zugreifen können. Eine solche integrierte Datenbasis war in Unternehmen in den siebziger Jahren weitgehend unbekannt. Die Folge waren enorme Kosten aufgrund redundanter und inkonsistenter Daten. Darauf aufbauend entwickelte SAP nach und nach Module für verschiedene Funktionsbereiche, wie etwa Rechnungswesen und Controlling, Logistik oder Personalwirtschaft. Diese Pakete waren zunächst primär für industrielle Unternehmen konzipiert und konnten einzeln oder auch gemeinsam zusammen mit einer Datenbank gekauft werden.

Im Gegensatz zu Office-Anwendungen, wie sie etwa von Microsoft angeboten werden, zeigte sich, dass viele der betriebswirtschaftlichen Bereiche branchenspezi-

fische Besonderheiten aufweisen. Daher wird von der SAP mittlerweile eine Vielzahl von Branchenlösungen angeboten, auch für den zunächst weniger beachteten Dienstleistungssektor.

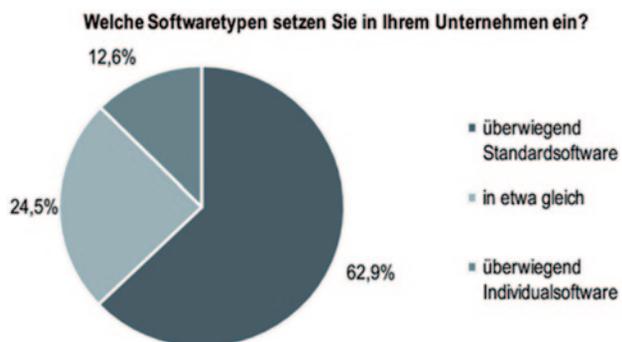
Echtzeitverarbeitung von Daten

Bis Ende der siebziger Jahre war es in Unternehmen üblich, mit Lochkarten zu arbeiten, d. h., die Daten wurden über Lochkarten in den Computer eingelesen und erst später verarbeitet. Die Idee der Echtzeitverarbeitung war bei den Gründern von SAP von Beginn an ein Kernaspekt und wurde in allen Systemen umgesetzt. Dies erklärt auch das Kürzel „R“ in den Produktnamen von SAP, das für „Realtime“ steht (Leimbach 2007; www.sap.com).

Die Grenze zwischen Individual- und Standardsoftware ist fließend. So kann betriebswirtschaftliche Standardsoftware zumindest bis zu einem gewissen Grad auch an die individuellen Bedürfnisse der Anwender angepasst werden. Allerdings verschlingen Einführungsprojekte, die eine Parametrisierung bzw. ein Customizing sowie gegebenenfalls eine Erweiterung der Software umfassen, häufig Millionenbeträge. Daher ist es für den Anwender in der Regel sinnvoll, lediglich kleinere Anpassungen vorzunehmen, zumal eine komplette Veränderung der Standardsoftware, bis sie 100 % auf seine Bedürfnisse zugeschnitten ist, zudem zu Problemen beim Umstieg auf Folgeversionen führen kann. Darüber hinaus bieten neuere Ansätze, wie serviceorientierte Architekturen, dem Anwender zumindest grundsätzlich die Möglichkeit, sich die besten Softwarepakete zu Teilbereichen herauszusuchen, diese auf seine Bedürfnisse anzupassen und mit Hilfe einer Integrationssoftware zu einer individuellen Anwendungslösung zusammenzuführen.

Zukünftig wird der Anteil von Standardsoftwarelösungen im Portfolio der Unternehmen tendenziell zunehmen, wie eine von uns durchgeführte Umfrage unter 498 deutschen CIOs zeigt (Buxmann et al. 2010). So nutzen 62,9 % der befragten Unternehmen überwiegend Standardsoftwarelösungen, in 24,5 % der Fälle setzen Unternehmen Standard- und Individualsoftware in etwa zu gleichen Teilen ein und bei 12,6 % der antwortenden Unternehmen wird überwiegend Individualsoftware eingesetzt (siehe Abb. 1.1).

Abb. 1.1 Anteile Softwaretypent



Neben der Erhebung des Status quo zur Nutzung von Standard- und Individualsoftware wurden die Unternehmen auch gefragt, wie sie die zukünftige Entwicklung bezüglich des Einsatzes dieser Softwaretypen einschätzen. 66,7% der Befragten stimmten der Aussage zu, dass in Unternehmen zukünftig tendenziell mehr Standardsoftware zu Lasten von Individualsoftware eingesetzt werden wird. 20,2% der Unternehmen sprachen sich gegen diese Aussage aus. 13,1% äußerten sich indifferent (siehe Abb. 1.2). 34,0% der Teilnehmer stimmten in diesem Zusammenhang der Aussage zu, dass zukünftig individuelle Anpassungen von Standardsoftware zurückgehen werden, knapp die Hälfte (49,0%) lehnte diese These dagegen ab. Der weitaus größte Teil (71,2%) der antwortenden Unternehmen ist der Auffassung, dass es aufgrund wachsender Anforderungen der Anwender immer wieder individuelle Lösungen für spezielle Problemstellungen geben wird, die in die gesamte IT-Landschaft zu integrieren sind.



Abb. 1.2 Einschätzung der zukünftigen Entwicklung für die Anteile der Softwaretypen

Zukünftig werden Unternehmen somit mehr Standardsoftwarelösungen einsetzen; allerdings werden weiterhin individuelle Anpassungen notwendig sein. Für Spezialprobleme, für die es (noch) keine standardisierte Anwendungssoftware auf dem Markt gibt, wird man nach Maßgabe von Wirtschaftlichkeitsüberlegungen tendenziell nach wie vor Individualsoftware entwickeln oder entwickeln lassen, die in die IT-Landschaft der Unternehmen einzubinden ist.

Neben Softwareanbietern im engeren Sinne – also den Unternehmen, die Standard- oder Individualsoftware entwickeln – gibt es Anbieter für Dienstleistungen bzw. Services in den späteren Phasen im Lebenszyklus einer Softwarelösung. Wir bezeichnen diese Unternehmen im Folgenden als Softwareanbieter im weiteren Sinne. Diese Dienstleistungen umfassen zum einen die Unterstützung von Anwendern bei der Implementierung der Softwarelösungen sowie zum anderen den Betrieb der Produkte. Abbildung 1.3 fasst unsere Klassifikation von Softwareanbietern zusammen.

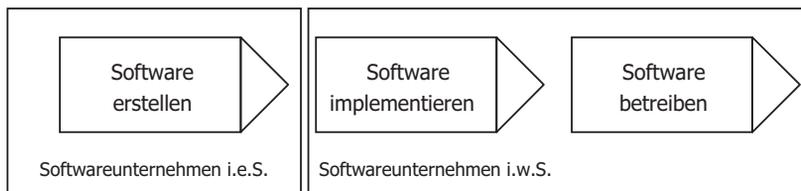


Abb. 1.3 Klassifikation von Softwareunternehmen

Insbesondere für erklärungsbedürftige Softwarelösungen, also solche, die sich nicht leicht implementieren und in ein Anwendungsumfeld integrieren lassen, existiert eine enorme Nachfrage nach Dienstleistungen. Demzufolge gibt es eine Vielzahl von Anbietern auf diesem Markt. Dabei handelt es sich um (Lünendonk 2009).

- IT-Beratungs- und Systemintegrations-Unternehmen (mit Angeboten im Bereich IT-Beratung, Entwicklung von Individualsoftware etc.),
- IT-Serviceunternehmen (mit Leistungen wie Outsourcing, ASP, Schulung etc.),
- Business Innovation/Transformation Partner (Management- und IT-Beratung sowie System-Realisierung).

Eine besonders große Rolle spielt traditionell die Unterstützung der Anwender bei SAP-Einführungsprojekten. Dies liegt daran, dass – wie bereits angesprochen – die Software im Rahmen dieser Projekte an die speziellen Bedürfnisse der Anwender angepasst wird. Entsprechende Kenntnisse liegen in den Anwendungsunternehmen in vielen Fällen nicht vor oder es stehen dort nicht genügend Mitarbeiter mit einem solchen Know-how zur Verfügung. Diese Einführungsprojekte sind grundlegend für den Nutzen, den die Software für die Kunden stiftet, da sie direkt in die inner- sowie zwischenbetrieblichen Prozesse eingreift. In diesem Zusammenhang sind häufig auch kleinere Programmierarbeiten, etwa die Entwicklung von Schnittstellen zwischen heterogenen Systemen, zu erledigen.

Dabei setzt dieses Geschäft wie kaum ein anderes eine Vertrauensbasis zwischen den Kunden und den Dienstleistungsanbietern voraus. Hoch et al. (2000, S. 160) benutzen in diesem Zusammenhang sogar den Begriff des „Glaubens“: Der Kunde muss daran glauben, dass das Dienstleistungsunternehmen bzw. die für das Projekt eingesetzten Mitarbeiter ein hervorragendes Know-how haben und in der Lage sind, ihr Versprechen zu halten, die Probleme der Kunden zu lösen. Denn während ein Anwender die Möglichkeit hat, die Software eines Standardsoftwareherstellers zu testen, ist dies im Beratungsgeschäft nicht möglich. Dem Kunden bleibt häufig nichts anderes übrig, als seine Entscheidung auf die Referenzen der Beratungshäuser und Systemintegratoren zu stützen. Daher ist es auch das vorrangige Ziel der Marketingaktivitäten von diesen Dienstleistungsunternehmen, Vertrauen in ihre Kompetenz aufzubauen. Hierzu gehören insbesondere die folgenden Maßnahmen (Hoch et al. 2000, S. 162–178):

- Sponsoring von IT-Konferenzen,
- Diskussionszirkel mit hochrangigen Vertretern der IT- und Softwareindustrie,
- Veröffentlichungen in Fachmagazinen und wissenschaftlichen Zeitschriften sowie
- Anzeigen und TV-Spots.

Aufgrund der Komplexität von Einführungs- und Integrationsprojekten ist die Auswahl des Dienstleistungsunternehmens für den Kunden von zentraler Bedeutung. Dies gilt insbesondere auch, weil in den meisten Fällen das Budget überschritten wird bzw. viele dieser Projekte komplett scheitern. So wurden nach einer Untersuchung der Standish Group International, Inc. lediglich 32% der betrachteten IT-Projekte innerhalb der Zeit- und Budgetvorgaben erfolgreich abgeschlossen (The Standish Group International, Inc. 2009, S. 2). 44% überschritten diese Planwerte erheblich und 24% wurden überhaupt nicht zu Ende geführt. Bei den als komplex eingestuften Projekten wurden Durchlaufzeit und Budget um mehr als das Doppelte überschritten.

Der Betrieb von IT-Lösungen gehört seit Jahren zum Outsourcing-Geschäft (siehe hierzu auch Kap. 5). Hier gibt es eine Vielzahl von Playern in einem weltweit umkämpften Markt, die wir auch zu den Softwareanbietern i. w. S. zählen.

1.3.2 Die Auswahl von Software

Grundlage der Entwicklung und Gestaltung von Geschäftsmodellen für Softwareanbieter im engeren sowie im weiteren Sinne sind die Auswahlentscheidungen der Kunden. Die folgenden Abschnitte sollen aufzeigen, welche Prozesse und Entscheidungsfaktoren bei der Auswahl und Bewertung alternativer Softwarelösungen in Unternehmen (Abschn. 1.3.2.1) und Haushalten (Abschn. 1.3.2.2) existieren.

1.3.2.1 Softwareauswahl in Unternehmen

In der Regel durchlaufen Unternehmen mehrere sequenzielle Phasen bis zur endgültigen Auswahlentscheidung (siehe Abb. 1.4). Hierbei wird der Lösungsraum der zur Verfügung stehenden Alternativen schrittweise eingegrenzt. Typische Kontextvariablen sind Unternehmensaspekte, wie z. B. Branchenzugehörigkeit, Größe oder zu unterstützende Funktionen, bzw. Umfeldvariablen, wie etwa Technologiestandards. Aus den Rahmenbedingungen heraus werden Ziele definiert, die mit der Anschaffung der Software verfolgt werden sollen. Der nächste Schritt stellt in der Regel die Definition von Kriterien dar. Kriterien sind relevante Merkmale für die Beschreibung eines Softwaresystems und die Grundlage für eine Bewertung.



Abb. 1.4 Softwareauswahl in Unternehmen

Mit Hilfe von Marktübersichten werden potenzielle Alternativen ermittelt. Typisch ist das sich anschließende zweistufige Bewertungs- und Auswahlverfahren. Bei der Grobauswahl werden die Alternativen anhand so genannter K.O.-Kriterien bewertet. Ihre Erfüllung ist notwendige Voraussetzung für eine detailliertere Untersuchung. Für die verbliebenen Alternativen werden auf Grundlage detaillierter Anforderungsdefinitionen (Pflichtenhefte), die aus den Kriterien abgeleitet werden, entsprechende Angebote eingeholt bzw. Informationen angefordert. Basierend auf den auf diese Weise gewonnenen Informationen wird eine Feinauswahl vorgenommen, die in einen Entscheidungsvorschlag mündet. Einige Ansätze berücksichtigen ferner die an die Entscheidung anschließenden Phasen, wie Vertragsverhandlungen und funktionale Testläufe zur Bestimmung des Anpassungsaufwandes.

Im Folgenden soll detaillierter auf die Entwicklung eines Zielsystems, die Kriterienableitung und -gewichtung sowie auf die während der Grob- und Feinauswahl relevanten Aktivitäten eingegangen werden.

Die Formulierung der Ziele, die mit der Einführung bzw. dem Einsatz einer Software in einem bestimmten Bereich erreicht werden sollen, stellt in der Regel den Ausgangspunkt der Analyse und den Maßstab zur Beurteilung der Alternativen dar. Bei der Ableitung der Ziele können unterschiedliche Verfahren zum Einsatz kommen: Zum einen werden Ziele top-down aus den obersten Unternehmenszielen bzw. aus der IT-Strategie abgeleitet. Das kann in der Weise geschehen, dass oberste Unternehmensziele, wie beispielsweise die Gewinnsteigerung, auf das konkrete Entscheidungsproblem „Auswahl einer Standardsoftware für ...“ übertragen werden. Eine Variante dieser Zielableitung stellt eine Konkretisierung der obersten Ziele durch Unterziele im Sinne einer Zweck-Mittel-Beziehung bezogen auf das Entscheidungsproblem dar. Beispiele hierfür sind die Senkung der Durchlaufzeit, die Senkung von Lagerbeständen oder die Verkürzung von Reaktionszeiten.

Ein anderer Ansatz geht entgegengesetzt zur oben vorgestellten Strategie bottom-up vor. Diese Ziele haben einen stark inhaltlich-funktionalen Fokus mit Bezug zum konkreten Entscheidungsproblem bzw. zu existierenden Strukturen (wie z. B. die bestehende IT-Architektur). Beispiele sind Ziele in Form allgemeiner Softwaremerkmale, wie Datenbank- oder Hardwareherstellerunabhängigkeit, Kompatibilität mit bestehenden Versionen oder spezifische anwendungsbezogene Funktionalitäten, wie beispielsweise eine automatische Teilenummernvergabe mit Prüfziffer oder die Verbreiterung der Informationsbasis. Diese Aspekte stellen oftmals eher bereits Anforderungen dar und nicht Ziele im engeren Sinn, die um ihrer selbst willen verfolgt werden. In der Praxis wird deshalb meist ein kombiniertes Verfahren aus top-down und bottom-up-Vorgehensweise zur Zielbestimmung als Voraussetzung der Kriterienableitung angewendet. Allerdings wird auf das erforderliche Gegenstromprinzip im Sinne eines Abgleichens und Anpassens oftmals zugunsten einer vereinfachten Aufzählung verzichtet.

Werden Kriterien aus dem Zielsystem abgeleitet, so stellen sie in der Regel die unterste Ebene eines solchen Systems dar. Die Auswirkung einer bestimmten Systemeigenschaft auf die angestrebten Ziele ist damit transparent. Im Sinne der Zielhierarchie werden die Auswahlkriterien häufig in software-, implementierungs- und anbieterbezogene Merkmale

untergliedert. Neben den eigentlichen Leistungsmerkmalen der Software spielen bei komplexeren Systemen insbesondere Implementierungskriterien eine wesentliche Rolle, da sie neben den Anschaffungskosten erhebliche Auswirkungen auf die Gesamtaufwendungen für die Software haben können. Die Implementierung und Einführung komplexerer Software-Systeme werden oftmals von Softwareanbietern bzw. von Implementierungspartnern begleitet, deren Expertise und Professionalität im Vorfeld der Softwareauswahl in der Regel auch zu evaluieren ist. Neben rein rationalen Kriterien spielen darüber hinaus in der Praxis auch häufig politische Kriterien (z. B. Absprachen, Klüngelei, Machtspiele) eine nicht zu unterschätzende Rolle, die den Auswahlprozess erheblich beeinflussen können (Howcroft und Light 2006).

Tabelle 1.1 zeigt für das Beispiel der Auswahl von ERP-Systemen, welche konkreten Auswahlkriterien für die Dimensionen „Software“, „Implementierung“ und „Anbieter“ in der Praxis herangezogen werden (Keil und Tiwana 2006; Jadhav und Sonar 2009).

Tab. 1.1 Typische Auswahlkriterien bei der Auswahl von ERP-Systemen

Auswahlkriterien (Software)	Auswahlkriterien (Implementierung/Anbieter)
Funktionalität	Integrationsfähigkeit in bestehende IT-Architektur
Kosten	Implementierungszeit/Anpassungsaufwand
Benutzerfreundlichkeit	Support durch Anbieter
Zuverlässigkeit	Reputation des Anbieters

Hinsichtlich der Gewichtung der Einzelkriterien im Rahmen der Gesamtentscheidung wird in der Regel davon ausgegangen, dass die Kriterien vorhanden bzw. erfüllt sein sollen. Andererseits ist auch denkbar, dass die Erfüllung eines Kriteriums in Abstufungen möglich ist. In einem solchen Fall ist die Angabe eines angestrebten Zielniveaus bzw. die Beurteilung von unterschiedlichen Zielniveaus erforderlich. So ist allein aus einem Kriterium „Datenbankunabhängigkeit“ nicht ersichtlich, ob hierzu die Unterstützung von zwei, drei oder zehn unterschiedlichen Datenbankmanagementsystemen erforderlich ist. Dies ist insbesondere zur Vermeidung von Überbewertungen von Bedeutung, wenn etwa bezüglich eines Ziels lediglich ein satisfizierendes Zielniveau angestrebt wird. Auf der anderen Seite liegen häufig auch Ziele vor, bei denen ein bestimmtes Anspruchsniveau mindestens erreicht werden muss, damit eine Alternative überhaupt in Betracht gezogen bzw. detaillierter untersucht wird. Solche K.O.- oder Killer-Kriterien tragen zwar einerseits dazu bei, den Auswahlprozess effizient zu gestalten. Andererseits bringen sie aber auch die Gefahr mit sich, dass Alternativen aufgrund einer auch nur geringen Unterschreitung des Mindestanspruchsniveaus eines Ziels ausgesondert werden.

Zur Gewichtung der Kriterien wird typischerweise ein holistischer Ansatz angewendet, wie er beispielsweise aus der Nutzwertanalyse bekannt ist. Dabei erfolgt die Vergabe der Gewichte durch einen direkten Vergleich der Kriterien. So wird z. B. in empirischen Studien zur Auswahl von ERP-Systemen berichtet, dass die Kriterien Funktionalität, Zuver-

lässigkeit und Kosten am meisten ins Gewicht fallen, während die Benutzerfreundlichkeit und der Implementierungsaufwand als weniger wichtig bewertet werden (Keil und Tiwana 2006). Bei diesem Ansatz erscheint die Vergabe sehr differenzierter Gewichtungen jedoch problematisch. Insbesondere bei komplexen Problemen, zu denen die Softwareauswahl u. a. aufgrund der großen Anzahl von Bewertungskriterien gehört, besteht allenfalls die Möglichkeit der Rangfolgenbildung in ordinaler Skalierung. Exaktere Angaben unterstellen (Schein-) Genauigkeiten, die in den seltensten Fällen tatsächlich die Präferenz des Entscheiders wiedergeben.

Marktüberblick, Alternativenvorauswahl und -bewertung vollziehen sich in der Praxis häufig interdependent und miteinander vernetzt. Schon die Berücksichtigung bzw. die Nichtberücksichtigung eines Systems beim Marktüberblick stellt eine Auswahl dar. Dieser erste Auswahlsschritt ist aufgrund der großen Marktbreite geprägt durch heuristische Ansätze, wie z. B.:

- Berücksichtigung von Systemen, die die Mitbewerber einsetzen,
- Berücksichtigung von Systemen, die gegenwärtig in der Fachpresse große Beachtung finden (z. B. Google Apps Premier, SAP ERP u. a.),
- Berücksichtigung von Systemen, mit denen Mitarbeiter bereits Erfahrungen haben,
- Berücksichtigung von Systemen, bei deren Auswahl sich ein Berater einen Anschlussauftrag im Rahmen der Implementierung erhofft,
- Berücksichtigung von Systemen aufgrund einer zufälligen Auswahl, z. B. durch Messebesuche.

Wie bereits dargestellt, bilden Bewertung und Auswahl einen gekoppelten Prozess, wobei die Anzahl der Alternativen mittels K.O.-Kriterien verringert werden kann. Dabei werden typischerweise die folgenden Verfahren angewendet:

- Auswertung von Systembeschreibungen,
- Auswertung bearbeiteter Ausschreibungsunterlagen bzw. Anforderungskataloge durch die potenziellen Anbieter sowie
- Präsentationen des Systems bzw. konkrete Funktionspräsentation auf Basis definierter Anwendungsszenarien.

Bei der Bewertung bestimmter Kriterien wird dabei häufig eine dreistufige Skala mit den Ausprägungen „erfüllt“, „teilweise erfüllt“, „nicht erfüllt“ zugrunde gelegt. Schließlich besteht das Bedürfnis, die Ergebnisse der Einzelbewertung zu einer übergreifenden Bewertung zusammenzufassen.

Da die Eignung eines Systems nicht nur aus seiner funktionalen Übereinstimmung mit den gestellten Anforderungen resultiert, sondern auch von den Kosten des Systems abhängt, werden diese häufig in der Weise in die Bewertung einbezogen, dass ein Kosten-Nutzen-Koeffizient gebildet wird. Auf Basis des Vergleichs entsprechender quantitativer und qualitativer Bewertungen wird schließlich die Auswahlentscheidung getroffen.

1.3.2.2 Softwareauswahl durch Konsumenten

Während es sich bei Auswahlentscheidungen in Unternehmen in der Regel um Gruppenentscheidungen handelt, werden Konsumentenentscheidungen meistens alleine getroffen. Ähnlich wie bei Auswahlentscheidungen im Unternehmenskontext durchläuft ein Konsument bei der Auswahl von Produkten im Allgemeinen und Software im Speziellen typischerweise vier Phasen (Blackwell et al. 2003), die in Abb. 1.5 illustriert werden.

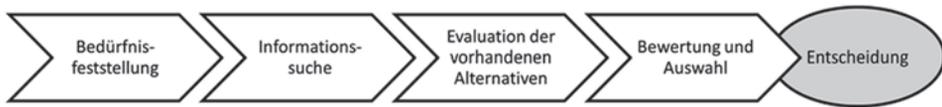


Abb. 1.5 Prozess zur Auswahl von Software durch Konsumenten

Nachdem ein Konsument sich des Bedürfnisses bewusst wird, eine neue Software kaufen zu wollen (Phase 1: Bedürfnisfeststellung), sucht er anschließend nach geeigneten Informationen über alternative Produkte, die das Bedürfnis befriedigen sollen. Dabei identifiziert der Konsument zunächst – mehr oder weniger bewusst – wesentliche subjektive und objektive Kriterien für seine Entscheidungsfindung (Phase 2: Informationssuche). Nach der Informationssammlung verarbeitet der Konsument die vorhandenen Daten und evaluiert jede Alternative vor dem Hintergrund seiner persönlichen Zielkriterien (Phase 3: Evaluation der Alternativen). Schließlich vergleicht und bewertet der Konsument die vorhandenen Alternativen und führt daraufhin eine Auswahl- bzw. Kaufentscheidung durch (Phase 4: Bewertung und Auswahl).

Im Gegensatz zu Auswahlentscheidungen im Unternehmen spielen bei Auswahlentscheidungen von Konsumenten neben rein objektiven Kriterien (wie Kosten oder Funktionalität einer Software) insbesondere individuelle bzw. interne (wie z. B. die Persönlichkeit, Emotionen, Wahrnehmungen oder der Lebensstil des Konsumenten) und situative bzw. externe Faktoren (z. B. Kultur oder Mund-zu-Mund-Propaganda) eine Rolle. Um aus Anbietersicht besser verstehen zu können, welche wesentlichen Kriterien Konsumenten bei der Software-Auswahl heranziehen, sollen diese nachfolgend vertiefend beleuchtet werden.

Wie oben bereits angeführt, spielen bei Konsumentenentscheidungen neben rein rationalen Auswahlkriterien insbesondere weichere Faktoren eine Rolle. Ein umfassendes Modell, das diesen Besonderheiten speziell in digitalen Produktwelten Rechnung trägt, ist das Modell der Erfahrungswerte digitaler Güter nach Mathwick et al. (2001; siehe Abb. 1.6). Es veranschaulicht, dass digitale Produkte (wie z. B. Spielesoftware) für Konsumenten einen Erfahrungswert besitzen, der vor dem eigentlichen Kauf (bewusst oder unbewusst) evaluiert wird. Der Erfahrungswert setzt sich dabei aus utilitaristischen und hedonischen Merkmalen zusammen.

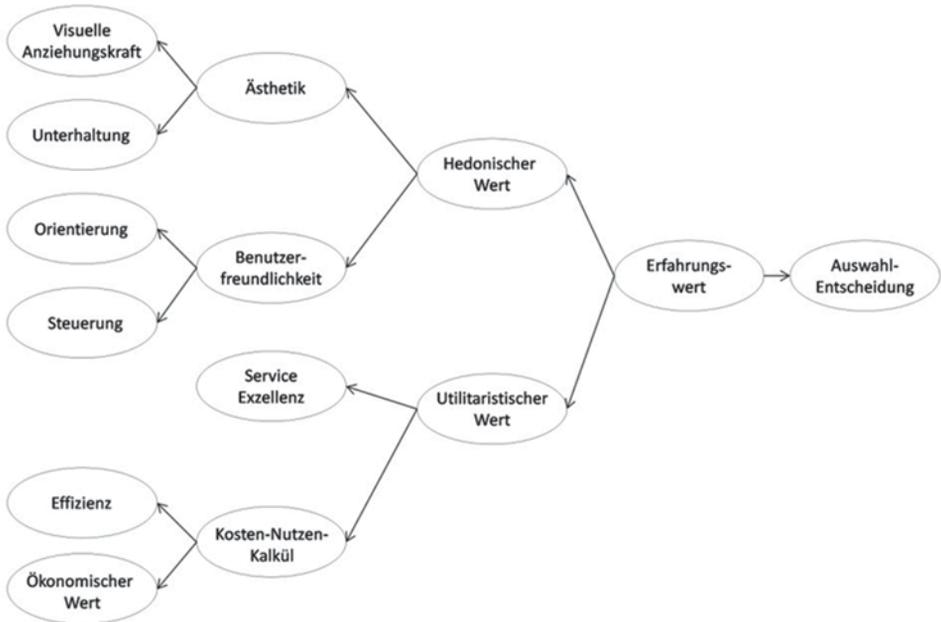


Abb. 1.6 Einflussfaktoren des Erfahrungswertes digitaler Güter. (in Anlehnung an Mathwick et al. (2001))

Utilitaristische Merkmale sind überwiegend objektiver Natur und entsprechen typischerweise Kosten-Nutzen-Kalkülen. In die Kosten-Nutzen-Betrachtung fließen wiederum weitgehend rationale Bewertungskriterien, wie z. B. die Funktionalität, Kompatibilität oder die Anschaffungskosten der Software, ein. Der mit der Software einhergehende Service beschreibt auf der anderen Seite, welche Unterstützungs- und Garantieleistungen (z. B. Hotline oder Gewährleistungsansprüche) der Nutzer beim Erwerb der Software erhält.

Hedonische Merkmale sprechen dagegen die oben angeführten persönlichen und situativen Bedürfnisse von Konsumenten an. So spielen ästhetische Vorlieben von Konsumenten bei der Auswahl eine große Rolle wie auch die Benutzerfreundlichkeit der Software. Beide Faktoren umfassen konkretere Subkriterien, wie z. B. die visuelle Anziehungskraft (etwa Farbgestaltung, 3D-Effekte) oder die Art und Weise der Steuerung der Software (z. B. mit Maus, Tastatur, Controller), die wiederum spezifische, individuelle Kaufmotive des Konsumenten ansprechen. Insgesamt lassen sich über die Beeinflussung der Subkriterien die darüber liegenden hedonischen und utilitaristischen Werte und somit der gesamte Erfahrungswert einer Software beeinflussen, der wiederum einen Einfluss auf die Software-Auswahl und Kaufentscheidung des Konsumenten ausübt. Die Subkriterien lassen sich in diesem Sinne mehr oder weniger stark vonseiten eines Softwareanbieters beeinflussen.

1.4 Dienstleistungserlöse in der Softwareindustrie

Neben den Lizenzeeinnahmen erzielen Softwareanbieter in zunehmendem Maße Dienstleistungserlöse. Diese Erlösquelle ist dabei nicht allein den Individualsoftwareherstellern sowie den Softwareanbietern im weiteren Sinne vorbehalten. Auch für Standardsoftwarehersteller bieten sich hier verschiedene Möglichkeiten, Umsätze zu generieren, etwa indem sie selbst Services anbieten, wie z. B. Beratungs- sowie Wartungsdienstleistungen.

Während die Beratungsdienstleistungen meistens nach Beratertagen (und gelegentlich auch erfolgsabhängig) abgerechnet werden, sehen die Erlösmodelle für den Bereich Wartungsservices in der Regel vor, dass der Anwender jährlich einen Prozentsatz der Lizenzgebühren an den Softwareanbieter zahlt. Dieser Prozentsatz variiert von Anbieter zu Anbieter – eine typische Größe liegt bei ca. 20% pro Jahr. Damit wird unmittelbar deutlich, von welcher Bedeutung diese Erlösquelle für die Softwareanbieter ist: Wenn wir davon ausgehen, dass eine bestimmte Softwarelösung im Durchschnitt sieben bis zehn Jahre genutzt wird, bevor sie durch eine neue Version oder – seltener – durch ein Alternativprodukt ersetzt wird, zeigt sich zum einen, dass die Wartungserlöse die Lizenzeeinnahmen in der Regel übersteigen; dies gilt auch nach Diskontierung der Erlöse. Zum anderen ergibt sich aus diesem Geschäftsmodell auch, dass Wartungserlöse relativ konstant über den Zeitverlauf anfallen und – im Gegensatz zu den Lizenzeeinnahmen – eine relativ gut planbare Erlösquelle für die Softwareanbieter sind. Abbildung 1.7 zeigt, dass die meisten der großen Standardsoftwareanbieter ihren Anteil des Servicegeschäfts am Gesamtumsatz in den letzten Jahren recht konstant halten konnten.

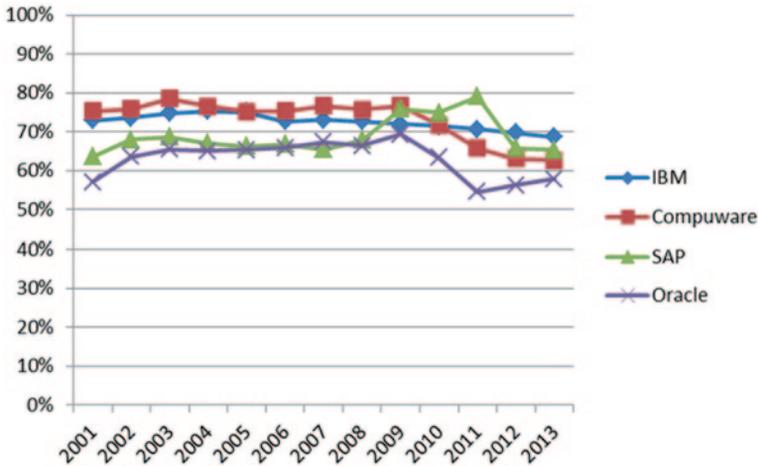


Abb. 1.7 Anteil des Servicegeschäfts am Umsatz großer Standardsoftwarehersteller. (erweiterte eigene Berechnungen auf Basis von Cusumano (2004, S. 37))

Dass die Serviceumsätze die Lizeinnahmen in vielen Fällen deutlich übersteigen, bedeutet jedoch nicht, dass das Servicegeschäft grundsätzlich attraktiver als der Verkauf von Lizenzen ist. Betrachten wir die Profitabilität der verschiedenen Erlösquellen, so liegt diese für das Lizenzgeschäft im Regelfall bei etwa 90–100% (Cusumano 2004, S. 4 f.). Variable Kosten, wie zum Beispiel für Datenträger, Handbücher, Dienstleistungsanbieter oder Verpackung, sind meist niedrig – beim Vertrieb über das Internet sind sie sogar vernachlässigbar gering. Demgegenüber liegen die Deckungsbeiträge für Beratungs- und Wartungsdienstleistungen deutlich darunter.

Von großer Bedeutung ist die Profitabilität der unterschiedlichen Geschäftsbereiche für Investoren, die sich in vielen Fällen eher an Renditen als an absoluten Werten orientieren. Damit sind Softwareanbieter, die sich überwiegend auf den Produkt- bzw. Lizenzverkauf konzentrieren, häufig für Investoren attraktiver als Softwareanbieter im weiteren Sinne. Dies liegt insbesondere daran, dass reine Produkthanbieter schneller wachsen können als Unternehmen, die einen Großteil ihrer Umsätze mit Services erzielen. Daher haben viele Softwareunternehmen auch keine Beratungssparte. Andererseits haben Softwareanbieter, die einen großen Teil ihres Umsatzes mit Services erzielen, den Vorteil, dass sie – wie bereits oben erwähnt – relativ konstante Erlöse erzielen, was gerade in schlechteren wirtschaftlichen Zeiten mit wenig Neugeschäft von hoher Bedeutung ist.

Dabei stehen die in diesem Abschnitt untersuchten Lizenz- und Dienstleistungserlöse natürlich nicht unabhängig nebeneinander. Vielmehr werden die Höhe der Lizenzerlöse sowie die Preisstrategien der Anbieter die Dienstleistungseinnahmen maßgeblich mitbestimmen. So kann ein Geschäftsmodell in der Softwareindustrie etwa darin bestehen, Software zu niedrigen Preisen zu verkaufen oder gar zu verschenken. Dies ist häufig eine viel versprechende und zum Teil sogar notwendige Strategie, um frühzeitig Marktanteile zu gewinnen oder einem etablierten Wettbewerber Marktanteile abzunehmen. Unabhängig davon kann es auch interessant sein, komplementäre Dienstleistungen zu der günstig abgegebenen oder verschenkten Software anzubieten und die Umsätze ausschließlich oder überwiegend mit Services zu generieren.

Solche Geschäftsmodelle sind insbesondere für die Unternehmen anwendbar, die digitale Güter, wie etwa Software oder Musik, anbieten. Über die Erlösgenerierung hinaus hat das Gut Software einige weitere spezifische Eigenschaften, die es von anderen Produkten unterscheidet. Auf die ökonomischen Hintergründe und die Folgen für Softwaremärkte wollen wir im folgenden zweiten Kapitel eingehen.

Literatur

- Blackwell RD, Miniard PW, Engel JF (2003) Consumer behavior. Harcourt, Orlando
- Buxmann P, Brandt B, von Ahsen A, Hess T (2010) Outsourcing der Entwicklung und Anpassung von Anwendungssoftware: analyse der Kundenzufriedenheit auf Basis einer empirischen Untersuchung. Darmstädter Arbeitspapier
- Campbell-Kelly M (1995) Development and structure of the international software industry 1950–1990. *Bus Econ Hist* 24:73–110

- Cusumano MA (2004) *The business of software: what every manager, programmer and entrepreneur must know to succeed in good times and bad*. Simon & Schuster, New York
- Hoch DJ, Roeding CR, Purkert G, Lindner SK (2000) *Erfolgreiche Software-Unternehmen. Die Spielregeln der New Economy*. Hanser, München
- Howcroft D, Light B (2006) Reflections on issues of power in packaged software selection. *Inf Syst J* 16:215–235
- Ichbiah D (1993) *Die microsoft story: Bill Gates und das erfolgreichste Softwareunternehmen der Welt*. Campus, München
- Jadhav AS, Sonar RM (2009) Evaluating and selecting software packages: a review. *Inf Softw Technol* 51:555–563
- Keil M, Tiwana A (2006) Relative importance of evaluation criteria for enterprise systems: a conjoint study. *Inf Syst J* 16:237–262
- Leimbach T (2007) Vom Programmierbüro zum globalen Softwareproduzenten. Die Erfolgsfaktoren der SAP von der Gründung bis zum R/3-Boom. *Z Unternehmensgeschichte* 52:5–34
- Lünendonk T (2009) *Führende IT-Beratungs-, IT-Service- und Standard-Software-Unternehmen in Deutschland*. Bad Wörishofen
- Lünendonk T (2013) TOP 25 der Standard-Software-Unternehmen in Deutschland 2012. http://lunendonk.de/wp-content/uploads/2013/05/LUE_Liste_u_PI_2013_Standard_Software_f160520131.pdf
- Marly J (2014) *Praxishandbuch Softwarerecht*, 6 Aufl. Beck, München
- Mathwick C, Malhotra N, Rigdon E (2001) Experiential value: conceptualization, measurement and application in the catalog and internet shopping environment. *J Retailing* 77:39–56
- Mertens P, Große-Wilde J, Wilkens I (2005) *Die (Aus-)Wanderung der Softwareproduktion – Eine Zwischenbilanz*, Bd 38. Institut für Informatik der Friedrich-Alexander-Universität Erlangen-Nürnberg, Erlangen-Nürnberg
- The Standish Group International, Inc. (2009) *New Standish Group report shows more project failing and less successful projects*. http://www1.standishgroup.com/newsroom/chaos_2009.php. Zugegriffen: 23. April 2009, Massachusetts