

Institut für Wirtschaftsinformatik
Universität Hannover

Potenziale und Risiken von Standard- und Individualsoftware

VERFASSER:

Liina Vaher

BETREUER:

Dipl.-Ök. Andreas Priess

IWI-Seminar „Standard- und Individualsoftware: Einsatzmöglichkeiten, Vor- und Nachteile“

Wintersemester 2003/2004

Inhaltsverzeichnis

Abkürzungsverzeichnis	2
1 Einleitung	3
2 Grundlagen	3
2.1 Untergliederung der Software	3
2.2 Merkmale von Standardsoftware	5
2.3 Merkmale von Individualsoftware	6
2.4 Modifikation der Standardsoftware	7
3 Kriterien für die Auswahl und den Einsatz von Standard- bzw. Individualsoftware	8
3.1 Gegenüberstellung von Standardsoftware und Individualsoftware	8
3.2 Auswahl der Software unter Berücksichtigung verschiedener Einsatzbereiche .	11
3.2.1 Geschäftsprozesse	11
3.2.2 Abteilungen	12
3.2.3 Geschäftsformen	13
4 Ausnutzung der Potenziale und Vermeidung der Risiken von Standard- und Individualsoftware	15
4.1 Standardsoftware	15
4.2 Individualsoftware	16
5 Fazit	18
Literaturverzeichnis	20

Abkürzungsverzeichnis

ABAP	Advanced Business Application Programming
EDV	Elektronische Datenverarbeitung
ERP	Enterprise Ressource Planning
ISW	Individualsoftware
IT	Informationstechnologie
IV	Informationsverarbeitung
SSW	Standardsoftware
zfbf	Zeitschrift für betriebswirtschaftliche Forschung

1 Einleitung

In den letzten Jahren hat sich das Informationsmanagement immer mehr zu einem wettbewerbspolitischen Einsatzinstrument in vielen Unternehmen entwickelt. Im Zuge der wachsenden Internationalisierung und der damit einhergehenden Forderung nach mehr Flexibilität und Kundenorientierung, aber auch wegen der rasanten Entwicklung im Bereich der Technologie, wird immer häufiger nach einer besseren EDV-technischen Unterstützung der Unternehmenstätigkeiten gesucht. Alles, was zeitraubend, umständlich oder kostenintensiv ist, sollte möglichst mit Hilfe der Informationstechnologie verkürzt, vereinfacht und preiswerter werden. Für viele dieser Wünsche gibt es in der Softwarebranche bereits interessante Lösungsansätze.

Die Einführung von Software stellt eine grundlegende und weitreichende Entscheidung dar, weil sie das Unternehmen in der Regel für mehrere Jahre an sich bindet und damit die Funktionsweise und den Erfolg des Unternehmens erheblich mitbestimmt. Eine wichtige Entscheidung im Rahmen einer Softwareeinführung ist darin zu sehen, ob eine auf dem Markt befindliche Standardanwendungslösung oder aber eine eigenentwickelte Individualsoftware eingesetzt werden soll. Das Ziel dieser Arbeit ist zu erläutern, welche entscheidungsrelevanten Kriterien bei der Auswahl von Standard- bzw. Individualsoftware für ein Unternehmen eine Rolle spielen und wie die Potenziale und Risiken, welche die beiden Softwarearten mit sich bringen, auszunutzen bzw. zu vermeiden sind.

Dafür wird in Kapitel 2 eine Untergliederung der Software vorgenommen und die Merkmale der Softwarearten werden definiert und kurz erklärt, um die theoretischen Grundlagen zu schaffen. Weiterhin wird in Kapitel 3 die Entscheidung zwischen der Eigenerstellung und dem Kauf der Software anhand verschiedener Vor- und Nachteile der Standard- bzw. Individualsoftware diskutiert. Es wird ebenfalls erläutert, welche Auswirkungen die Potenziale und Risiken der beiden Softwarearten auf die Auswahl und den Einsatz bei verschiedenen Geschäftsformen, Geschäftsprozesse und Abteilungen haben. Kapitel 4 erklärt, wie die Potenziale und Risiken der Standard- und Individualsoftware zu nutzen bzw. zu vermeiden sind.

2 Grundlagen

2.1 Untergliederung der Software

In der Literatur werden verschiedene Definitionen des Begriffs Software verwendet. Hansen u. a. definieren Software als die Gesamtheit aller Programme, die für EDV-Systeme zur Verfü-

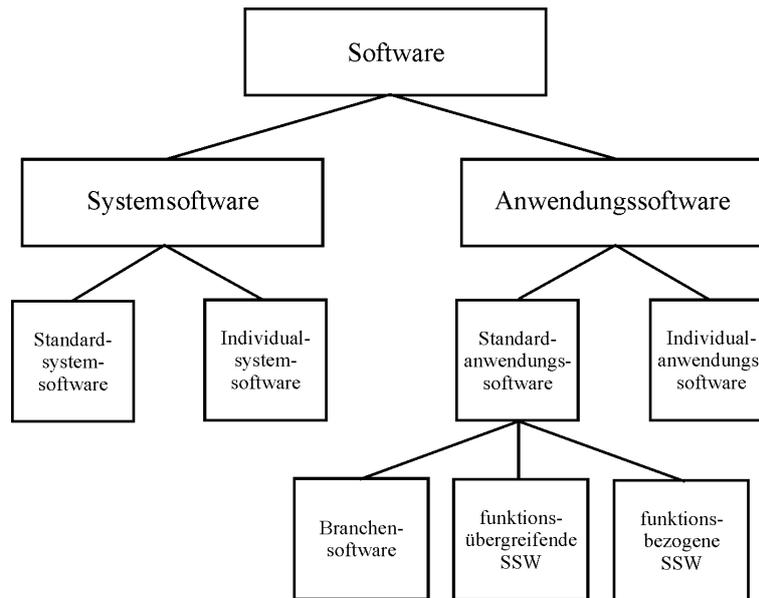


Abbildung 1: Softwareklassifizierung, eigene Darstellung in Anlehnung an [Diehl \[2000, S. 13\]](#); [Abts u. a. \[2002, S. 54\]](#)

gung stehen. Ein Programm wird dabei als die zur Lösung einer Aufgabe vollständige Anweisung an eine Datenverarbeitungsanlage gesehen.¹

Es werden bei der Software zwei Hauptgruppen unterschieden²: Systemsoftware und Anwendungssoftware (siehe Abb. 1). Systemsoftware ist erforderlich für den Betrieb und die Steuerung der Hardware, sowie für vielfältige Verwaltungs- und Überwachungsleistungen. Unter Anwendungssoftware werden diejenigen Programme verstanden, die zur Erfüllung, der vom Anwender gewünschten und veranlassten, Datenverarbeitungsaufgaben benutzt und zur Lösung betrieblicher Probleme eingesetzt werden.³

Weiterhin können System- und Anwendungssoftware in Individualsoftware (ISW) und Standardsoftware (SSW) klassifiziert werden.⁴ Zur Standardsoftware zählen Betriebs- und Datenbanksysteme, Netzwerksoftware und Softwareentwicklungssysteme. Die Individualsystemsoftware enthält Sonderfälle, wie z. B. eigenentwickelte Datenbanken.⁵ Ein typisches Beispiel einer Standardanwendungssoftware⁶ ist ein Buchhaltungsprogramm. Individualanwendungssoftware ist auf die speziellen Einsatzbedürfnisse des Anwenders ausgerichtet, ein Beispiel hierfür ist die Steuerung einer Gepäckbeförderungsanlage.⁷ Die Betrachtung dieser Arbeit liegt im Bereich der Anwendungssoftware. Daher wird im Folgenden unter Standard- und Individualsoftware immer Standard- bzw. Individualanwendungssoftware verstanden.

¹Vgl. [Hansen u. a. \[1983, S. 7\]](#).

²vgl. [Frank \[1977, S. 13–14\]](#); [Abts u. a. \[2002, S. 54–55\]](#); [Schlichtherle \[1998, S. 2–3\]](#); [Hansen u. a. \[1983, S. 7–9\]](#)

³Vgl. [Hansen u. a. \[1983, S. 9\]](#); [Mertens u. a. \[2000, S. 12\]](#); [Frank \[1977, S. 13–14\]](#); [Kirchmer \[1996, S. 13\]](#)

⁴Für eine Definition der SSW und ISW siehe Kapitel 2.2 und 2.3.

⁵Vgl. [Diehl \[2000, S. 13\]](#).

⁶Eine weitere Unterteilung findet im Kapitel 2.2 statt.

⁷Vgl. [Schlichtherle \[1998, S. 2\]](#); [Mertens u. a. \[2000, S. 13\]](#)

Da einige Softwaresysteme, wie z. B. Softwareentwicklungs- oder Datenbankmanagementsysteme, sich den oben genannten Kategorien nicht eindeutig zuordnen lassen, sollte diese Klassifizierung nicht als absolute Einordnung, sondern eher als Hilfe verstanden werden.⁸

2.2 Merkmale von Standardsoftware

Unter Standardsoftware werden vorgefertigte Programmpakete verstanden, die einen klar definierten Anwendungsbereich unterstützen und als fertige Produkte auf dem Markt erworben werden können.⁹ Spezifisch für SSW ist, dass sie auf Allgemeingültigkeit ausgelegt ist und damit in vielen verschiedenen Unternehmen einsetzbar ist, d. h. dass sie in Unternehmen unterschiedlicher Branchen sowie in Unternehmen mit unterschiedlichen Organisations- und Ablaufstrukturen verwendet werden kann.¹⁰

Grundsätzlich wird bei Standardsoftware zwischen Funktionssoftware und Branchensoftware unterschieden.¹¹ Funktionssoftware ist branchenneutral und für bestimmte Aufgabenbereiche zugeschnitten, die stark geregelt, und deswegen für Software mit hohem Standardisierungsgrad geeignet ist.¹² Wenn betriebliche Standardsoftware als einzelnes Programm für ein bestimmtes betriebliches Arbeitsgebiet erworben wird, dann spricht man von funktionsbezogener SSW. Ein Beispiel dafür ist Buchhaltungssoftware, die für die Automobilindustrie oder den Flugzeugbau geeignet ist. Funktionsübergreifend ist die SSW, wenn sie als geschlossenes (integriertes) Programmpaket für mehrere Anwendungsgebiete (Software-Familie) verwendbar ist. Ein Beispiel hierfür ist SAP R/3. Bei integrierten Softwarepaketen werden die Aufgaben so abgewickelt, dass keine redundanten Daten vorkommen. Die einzelnen Module solcher Programmpakete greifen dabei alle auf dieselbe Datenbank zu. Der Vorteil des modularen Aufbaus liegt darin, dass die Möglichkeit vorhanden ist, nur die Funktionalitäten zu implementieren, die zur Zeit wirklich benötigt werden.¹³

Obwohl Standardsoftware definitionsgemäß branchenneutral ist, haben sich dennoch verschiedene Anwendungen durchgesetzt, die auf die Anforderungen einzelner Wirtschaftsbereiche zugeschnitten sind. Diese werden Branchensoftware genannt (siehe Tabelle 1).¹⁴ Beispielsweise ist für Krankenhäuser branchenspezifisch, wie die Raum- bzw. Bettenreservierung und die Leistungsabrechnung mit den Versicherungsträgern abläuft.¹⁵ Der Einsatz der Branchensoftware gewinnt immer mehr an Bedeutung gegenüber der branchenneutralen SSW, da hierbei die

⁸Vgl. Schlichtherle [1998, S. 2].

⁹Vgl. Schwarze [1994, S. 279]; Abts u. a. [2002, S. 55]; Schwarzer u. a. [1996, S. 220–221]

¹⁰Vgl. Diehl [2000, S. 13].

¹¹Vgl. Schwarzer u. a. [1996, S. 224–225].

¹²Vgl. Böhm u. a. [1993, S. 126]; Schwarzer u. a. [1996, S. 221]

¹³Vgl. Schwarzer u. a. [1996, S. 224]; Stahlknecht u. a. [1999, S. 304]

¹⁴Vgl. Abts u. a. [2002, S. 67]; Stahlknecht u. a. [1999, S. 304]

¹⁵Vgl. Jochem [1997, S. 18].

Gesundheitswesen	Handwerk	Transport
Apotheken	Bäcker	Personenverkehr
Arztpraxen	Gärtner	Speditionen
Augenoptiker	Dachdecker	Taxibetriebe
Krankenhäuser	Maler	Containerdienste

Tabelle 1: Beispiele für Branchensoftware, in Anlehnung an [Schwarzer u. a. \[1996, S. 225\]](#); [Abts u. a. \[2002, S. 67\]](#)

individuellen Anforderungen der einzelnen Unternehmen stärker berücksichtigt werden können und deutlich geringere Kosten als bei einer ISW entstehen.¹⁶ Heute werden auch für große Unternehmen Branchensoftwarepakete angeboten. Beispiele sind sog. Industry Business Solutions von SAP, die z. B. für die Automobilindustrie, Luft- und Raumfahrt sowie Telekommunikation verfügbar sind.¹⁷ Da bei den Merkmalen und der Softwareauswahlentscheidung zwischen Standard- und Branchensoftware grundsätzlich keine Unterschiede bestehen, werden beide nachfolgend als Standardsoftware behandelt.

2.3 Merkmale von Individualsoftware

Individualsoftware wird für eine einzelne Unternehmung bzw. eine spezialisierte Aufgabenstellung entwickelt, d. h. es handelt sich um eine „maßgeschneiderte“ Software. Sie dient dazu, ein kundenindividuelles Problem zu lösen. Die Anforderungen einer Unternehmung können besser und gezielter mit ihrer Hilfe abgedeckt werden.¹⁸

Die Entwicklung von ISW kann als Eigenentwicklung von internen Mitarbeitern in der Informatikabteilung durchgeführt werden, aber auch eine Fremdentwicklung durch externe IT-Spezialisten ist möglich. Falls die Anwendungsprobleme wenig komplex sind, können die Anwender ihre Software mit Hilfe geeigneter Endnutzer-Tools selbst erstellen. Dabei sind auch Mischformen möglich.¹⁹

Um Individualentwicklungen zu realisieren, existieren viele verschiedene Entwicklungswerkzeuge und -sprachen, wie z. B. JAVA-basierte Werkzeuge, C++, Visual Basic usw.²⁰

Generelle Gründe für den Einsatz einer Individuallösung sind ein fehlendes Angebot von SSW auf dem Markt wegen begrenzter Nachfrage (z. B. Kalkulationssystem in der Mineralölindustrie) oder technische Rahmenbedingungen, die den Einsatz von ISW erforderlich machen. Ebenso kann eine strategische Überlegung, die einen Innovationsvorsprung und Wettbewerbsvorteil

¹⁶ Vgl. [Abts u. a. \[2002, S. 67\]](#).

¹⁷ Vgl. [Mertens u. a. \[2000, S. 28\]](#).

¹⁸ Vgl. [Diehl \[2000, S. 12\]](#); [Schwarzer u. a. \[1996, S. 222\]](#); [Jochem \[1997, S. 11\]](#)

¹⁹ Vgl. [Abts u. a. \[2002, S. 56–57\]](#); [Schwarzer u. a. \[1996, S. 222–223\]](#); [Mertens u. a. \[2000, S. 28\]](#)

²⁰ Vgl. [Bernhard u. a. \[1999, S. 2\]](#).

gegenüber den Konkurrenten bringt (z. B. Tarifmodelle bei Netzanbietern), ein Grund für eine Eigenentwicklung sein.²¹

2.4 Modifikation der Standardsoftware

In der Realität kann Standardsoftware oft nicht problemlos eingeführt werden und eine Individualentwicklung ist zu teuer und zu aufwendig. Die Lösung liegt deswegen in einem Mittelweg: im richtigen Mix aus Standard- und Individualsoftware. Es gibt verschiedene Möglichkeiten, den Mittelweg zu realisieren, die folgend erläutert werden.

Das Wort Standard bei der Standardsoftware bedeutet nicht unbedingt einen Verlust an Flexibilität und Individualität, da SSW sich unternehmensindividuell konfigurieren (auch Customizing genannt) und in gewissen Grenzen an die Wünsche des Anwenders anpassen lässt, um die große Diskrepanz zwischen den spezifischen betrieblichen Anforderungen und dem Funktionsumfang der SSW zu vermeiden. Zum Beispiel können die Menüs und Symbolleisten benutzerindividuell angepasst werden. Es können verschiedene Einstellungen an Standardsoftwareprodukten vorgenommen werden, so dass aus dem ursprünglichen „Standardprogramm“ ein „Maßanzug“ werden kann.²² Wie Schlichtherle sagt, „Ein ‚optimales‘ Customizing führt bei minimalen Mehrkosten zu einer angepassten Standardsoftware, die die Abläufe eines Unternehmens optimal unterstützt.“²³

Für die Anpassung der Standardsoftware an die individuellen Anforderungen können im Wesentlichen drei Möglichkeiten benutzt werden: Parametrisierung (Auswahl der Programmfunktionen durch Aufruf mit verschiedenen Parametern), Konfigurierung (auch Modularisierung genannt; Auswahl der gewünschten Programmbausteine) und Individualprogrammierung (individuelle Anpassung bzw. Ergänzung der Software).²⁴

Außerdem werden heutzutage immer mehr Standardsoftwaresysteme als offene Systeme konzipiert, was bedeutet, dass die Einbindung von Fremdprodukten (z. B. Tabellenkalkulationsprogramme, technische Systeme) oder eigener Entwicklungen über definierte Schnittstellen erfolgen kann. Damit können Änderungen und Erweiterungen an der ausgelieferten Form vorgenommen werden. Ein Beispiel hierfür ist SAP R/3, welches das Programmiersystem ABAP/4 enthält.²⁵

²¹ Vgl. Stahlknecht u. a. [1999, S. 303]; Schwarzer u. a. [1996, S. 222]; Abts u. a. [2002, S. 56]

²² Vgl. Schulz u. a. [2000, S. 38]; Mertens u. a. [2000, S. 147]

²³ Schlichtherle [1998, S. 152].

²⁴ Vgl. Stahlknecht u. a. [1999, S. 305–306].

²⁵ Vgl. Keil u. a. [1998, S. 857–858].

	Individualsoftware	Standardsoftware
Kosten		
Wartungs- & Serviceangebot		
Qualität/Stabilität		
Dokumentation		
Zeitaufwand		
Ergonomie		
Anpassungsfähigkeit		
Akzeptanz		
Einzigartigkeit		
Integration		
Belastung der Hardware		

Abbildung 2: Gegenüberstellung der Individual- und Standardsoftware, eigene Darstellung

3 Kriterien für die Auswahl und den Einsatz von Standard- bzw. Individualsoftware

3.1 Gegenüberstellung von Standardsoftware und Individualsoftware

Bei der Auswahl und Einführung eines neuen Softwaresystems in einer Unternehmung stellt sich die Frage, ob eine Standardlösung beschafft oder eine Individualsoftware entwickelt bzw. gekauft werden soll.²⁶ Um eine Entscheidung zu treffen, sind die Vor- und Nachteile der einzelnen Lösungen (siehe Abbildung 2), die in der Literatur viel Aufmerksamkeit gefunden haben²⁷, abzuwägen, damit die daraus resultierenden Potenziale und Risiken entdeckt werden können. Grundsätzlich gilt, dass die Vorteile der Standardsoftware den Nachteilen der Individualsoftware entsprechen und umgekehrt. Die wichtigsten Entscheidungsmerkmale werden nachfolgend erörtert.

Ein häufig genannter Vorteil der SSW ist, dass die Anschaffungskosten meistens niedriger als die Entwicklungskosten einer vergleichbaren ISW sind. SSW hat den Vorteil, dass die Entwicklungskosten auf mehrere Anwender verteilt werden können. Gleiches gilt auch für die Wartung und Schaffung aktueller Softwareversionen (Updates, neue Releases), die in der Regel verbesserte Funktionalität bringen und/oder Reaktionen auf Marktgegebenheiten sind. Auch hier kön-

²⁶Es wird in der Literatur auch häufig die Frage „Make or Buy“ genannt.

²⁷zur Diskussion der Vor- und Nachteile vgl. Schwarzer u. a. [1996, S. 221–224]; Schwarze [1994, S. 269]; Böhm u. a. [1993, S. 128–129]; Bernhard u. a. [1999, S. 2]; Jochem [1997, S. 12–13]

nen die Kosten auf eine Vielzahl von Anwendern aufgeteilt werden, was für eine Entscheidung für SSW spricht.²⁸ Noch ein Vorteil der SSW ist mit dem Wartungsservice verbunden: es steht eine Hotline mit Mitarbeitern, welche die Software genau kennen, für die Anwender zur Verfügung. In Fall der ISW gibt es weniger Detailwissen bei den Hotline-Mitarbeitern, falls überhaupt eine Hotline vorhanden ist.²⁹ Zusammengefasst bedeutet dies, dass ein potenzieller Kostenvorteil durch den Einsatz von SSW entsteht.

Auch die Qualität und Stabilität der ISW unterliegen meistens denen der SSW. Aufgrund der größeren Erfahrung der Programmierer des Anbieters einer SSW können die Fehler, die z. B. bei einer Installation aufgetreten sind, behoben und in Form eines Updates allen anderen Anwendern zur Verfügung gestellt werden.³⁰

Deutlich wird der Unterschied zwischen Eigenentwicklung und Standardsoftware bei der Dokumentation. Volumen und Qualität des Handbuchs der SSW übersteigen bei weitem die Möglichkeiten der ISW, wodurch die Einarbeitung bzw. Einführung der SSW potenziell leichter ist. Die Dokumentation der betriebsspezifischen Fachlösung bleibt oft in beiden Fällen ein Problem.³¹

Eines der gewichtigsten Argumente für SSW ist die Geschwindigkeit der Einführung. Die Entwicklungsdauer einer Individuallösung, um Softwaresysteme mit dem Leistungsumfang, Integrationsgrad und einer vergleichbaren Qualität einer SSW, die heute vom Markt bezogen werden kann, zu schaffen, ist zu hoch.³²

Bei SSW ist die Benutzeroberfläche häufig ergonomischer, wodurch die Einarbeitungszeit und der Schulungsbedarf der Mitarbeiter verringert wird. Es besteht aber das Risiko, dass die SSW überladen ist, da sie einen Funktionsumfang besitzt, den der Anwender nicht benötigt. Individuallösungen werden so entwickelt, dass ein spezieller Anwendungsfall abgedeckt wird.³³

Die Anpassungsfähigkeit der Software ist ebenfalls ein wichtiges Entscheidungskriterium. Einerseits bietet eine Individualentwicklung den Vorteil, dass keine Anpassungen an die Geschäftsprozesse des Unternehmens direkt nach dem Einsatz stattfinden müssen, da die Prozesse der Firma schon bei der Entwicklung der Software berücksichtigt werden können. Andererseits sind die Aufgaben und Abläufe nur selten auf längere Sicht starr, was eine Anpassungsfähigkeit der Software wichtig macht. Bei der SSW sind die Anpassungen an das Unternehmen bei der Einführung der Software möglich. Der Anwender kann die standardisierten Funktionen in einem eingeschränkten Umfang verändern, wobei die Grenzen durch die technischen Möglichkeiten der SSW gegeben sind. Die Anpassungen haben allerdings das Risiko, dass der Kosten-

²⁸ Vgl. Schlichtherle [1998, S. 150]; Schulz u. a. [2000, S. 46]

²⁹ Vgl. Schulz u. a. [2000, S. 47].

³⁰ Vgl. Schlichtherle [1998, S. 150]; Abts u. a. [2002, S. 74–76]

³¹ Vgl. Österle [1990, S. 26]; Arnold [1997, S. 64]

³² Vgl. Schulz u. a. [2000, S. 38].

³³ Vgl. Walther-Klaus [1999].

vorteil gegenüber einer ISW aufgebraucht wird, da in einigen Fällen die Investitionen in die Anpassung der SSW an das Unternehmen oder umgekehrt genau so hoch sein können wie die bei einer Eigenentwicklung.³⁴ Im Idealfall sind bei der ISW kurzfristig keine Anpassungen erforderlich. In der Realität haben sich aber zu Beginn der Entwicklung formulierte Anforderungen häufig geändert, da die Entwicklungszeiten bei der ISW relativ hoch sind. Spätere Anpassungen an sich veränderte Wettbewerbsbedingungen werden meist nicht systematisch geplant, wodurch ein Neuentwurf in vielen Fällen notwendig wird.³⁵

Für den Einsatz von ISW spricht die häufig bessere Akzeptanz des Systems durch die späteren Anwender, da bereits bei der Entwicklung eng mit den Mitarbeitern zusammengearbeitet werden kann, um die Bedienung des Systems an ihre Bedürfnisse anzupassen. Bei der Anschaffung von SSW besteht das Risiko, dass die Bereitschaft, sich mit der von außen übernommenen Lösung zu identifizieren und diese zu akzeptieren, nicht vorhanden ist.³⁶

Ein Argument, welches ebenfalls häufig gegen den Einsatz von SSW spricht, ist die Tatsache, dass sie in einer Unternehmung bestimmte Abläufe „erzwingt“. Dadurch wird die Abgrenzung zum Wettbewerber erschwert, sofern die Konkurrenten die gleiche SSW benutzen. Deshalb kann eine Speziallösung, die eine besondere Stärke eines Unternehmens aufbauen könnte, eine befriedigendere Lösung für die Unternehmung bieten. Außerdem hat ein einzelner Kunde einen geringen Einfluss auf die Entwicklungsrichtung von SSW.³⁷

Aus der Sicht der Organisation der Informationsverarbeitung ist das Vorhandensein bzw. Schaffen von Schnittstellen zu anderen Anwendungen bei der Entscheidung für SSW bzw. ISW wichtig. Die Einführung einer SSW in die bestehenden Datenmodelle der Unternehmung kann einerseits viel Arbeit verursachen. Andererseits ist der Integrationsgrad innerhalb einer SSW sehr hoch. Aktualität der Information, redundanzfreie Datenhaltung und bessere Kommunikation sind einige daraus resultierende Vorteile. Im Fall der Eigenentwicklung sind die Schnittstellen zu anderen Eigenentwicklungen bzw. Softwarepaketen anderer Anbieter zu schaffen, was einen großen Entwicklungsaufwand bedeutet. Es besteht das Risiko von Doppelentwicklungen grundlegender Problemlösungen, wodurch Kapazitäten ineffektiv gebunden werden. Auch Kompatibilitätsprobleme mit anderen Projekten sind möglich. Zusammenfassend spricht auch der Aspekt der Integration für den Einsatz von SSW.³⁸

Der Funktionsumfang, die Integration und die Flexibilität der SSW haben aber ihren Preis – die Beanspruchung der Computerressourcen ist hoch. Deshalb gilt im Allgemeinen, dass ISW geringere Hardwarekosten mit sich bringt.³⁹

³⁴ Vgl. Schwarze [1994, S. 271]; Abts u. a. [2002, S. 78–79]; Pietsch [1994, S. 6]

³⁵ Vgl. Keil u. a. [1998, S. 850–851].

³⁶ Vgl. Hagenlocher [2000, S. 61].

³⁷ Vgl. Schlichtherle [1998, S. 151]; Österle [1990, S. 21–23]

³⁸ Vgl. Österle [1990, S. 24–25]; Barbitsch [1996, S. 15]; Schwarzer u. a. [1996, S. 224]

³⁹ Vgl. Österle [1990, S. 28].

Aber nicht nur die Vor- und Nachteile der Softwarearten sind bei der Auswahl einer Softwarelösung entscheidend, auch die Bedürfnisse und Strukturen der Unternehmung, die Besonderheiten einzelner Abteilungen und Branchen, sowie der Funktionsumfang der Software spielen eine wichtige Rolle. Im weiteren Verlauf wird erläutert, welche Auswirkungen die Potenziale und Risiken der SSW und ISW auf die Softwareauswahl bei unterschiedlichen Geschäftsformen und -prozessen, sowie die Abteilungen haben.

3.2 Auswahl der Software unter Berücksichtigung verschiedener Einsatzbereiche

3.2.1 Geschäftsprozesse

Geschäftsprozesse sind definiert als eine abgeschlossene Abfolge von „logisch und sachlich zusammengehörenden Unternehmungsverrichtungen, die Ausgangsobjekte in Leistungen umsetzen“⁴⁰, und mit deren Hilfe die Bedürfnisse Dritter (z. B. Kunden) befriedigt werden können. Die Unterstützung der Geschäftsprozesse erfolgt meistens durch den Einsatz betriebswirtschaftlicher Anwendungssoftware.⁴¹ Die im Unternehmen eingesetzten EDV-Systeme sollen sich an den Geschäftsprozessen orientieren, sie dadurch unterstützen und somit die Ausführung der Unternehmensaktivitäten erleichtern. Da die geschäftsprozessorientierte Gestaltung der Organisationsstrukturen heutzutage eine zunehmende Bedeutung hat, wurde die funktionsorientierte Einführung der Software zu einem Problem, da die Schaffung der Schnittstellen zu anderen Bereichen schwierig wurde. Dies ist der Grund, warum heute in vielen Fällen die ERP-Systeme⁴² dominieren.⁴³

Die Anpassung betrieblicher Prozesse an die Software kann für veraltete Organisationsstrukturen vorteilhaft sein, wenn die Software zu effizienten Abläufen zwingt. Ein in vielen Unternehmen erprobtes und im Laufe der Zeit ausgereiftes Softwarepaket stellt normalerweise brauchbare Geschäftsprozesse zur Verfügung.⁴⁴ Es kann aber auch der Fall sein, dass eine Optimierung durch Standardsoftware nicht unterstützt wird, obwohl in gewissem Maße unterschiedliche Organisationsformen, Geschäftsabläufe und Strategien in SSW realisierbar sind. Das Risiko der Anpassbarkeit besteht. In diesem Fall stellt SSW ein Hindernis dar. Deswegen wird in der Literatur empfohlen, dass die Kernprozesse des Unternehmens – das sind die Prozesse, in denen die Kernkompetenz der Unternehmung begründet ist und in denen der eigentliche Gewinn erwirtschaftet wird – nicht durch Standardsoftware abgebildet werden sollen. Beispielsweise

⁴⁰Diehl [2000, S. 17].

⁴¹Vgl. Remme [1997, S. 29].

⁴²ERP-Systeme stehen für funktionsübergreifende betriebswirtschaftliche SSW, die zur Optimierung von Geschäftsprozessen eingesetzt werden, vgl. Enzler [2003, S. 1]

⁴³Vgl. Kirchmer [1996, S. 18–20].

⁴⁴Vgl. Pietsch [1994, S. 18].

kann die Planung der Produktionsabläufe ein Kernprozess der Firma sein, da es sicherlich nicht im Interesse der Unternehmung liegt, ihre Planungsstrategien offen zu legen und damit den Konkurrenten zu helfen. Auch der Kundenauftragsprozess eines Automobilherstellers, das Zahlungsverkehrssystem einer Bank oder das Buchungs- und Abwicklungssystem eines Touristikunternehmens sind besser mit ISW abzudecken. Einerseits um sich von den Konkurrenten zu differenzieren, andererseits kann in diesen Bereichen eine SSW fehlen, da die Anforderung zu speziell und die Einbindung in die Anwendungslandschaft zu komplex ist. Die unterstützenden Prozesse aber – das sind die Prozesse, die nicht unmittelbar zum Erfolg beitragen, trotzdem aber unverzichtbar sind, wenn das Unternehmen seine Aufgaben erfüllen soll – sind die idealen Kandidaten für SSW, da es sich bei ihnen in den meisten Fällen nicht lohnt, besser zu sein, als die Konkurrenten. Beispiele für unterstützende Prozesse sind die Finanzbuchhaltung und das Personalwesen.⁴⁵

3.2.2 Abteilungen

Bei der Betrachtung der verschiedenen Abteilungen innerhalb einer Unternehmung gilt im Allgemeinen, dass Standardsoftwaresysteme für gut verstandene, relativ allgemein gültige und abgrenzbare Problemstellungen und für wettbewerbsneutrale Bereiche geeignet sind. Beispiele sind der betriebswirtschaftliche Bereich (Finanzbuchhaltung, Kostenrechnung und Controlling) und die Personalwirtschaft (Administration und Disposition).⁴⁶ Es macht volkswirtschaftlich keinen Sinn, z. B. eine Personalabrechnung mehrfach neu zu entwickeln, wenn der Gesetzgeber bestimmte Regel festgelegt hat und dadurch keine Wettbewerbsvorteile gegenüber den Konkurrenten zu gewinnen sind.⁴⁷

Für die meisten Unternehmensbereiche wird heute leistungsfähige Standardsoftware angeboten: von der Personalwirtschaft, den Finanzen, dem Einkauf, der Produktion bis zum Vertrieb. Erfolgreiche Beispiele sind SAP R/3, Triton von Baan oder Oracle Applications.⁴⁸

Anders sieht es mit der Software für den Bereich Logistik aus. Hier lassen sich Wettbewerbsvorteile erzielen, welches auch der Grund für wenige SSW-Installationen in diesem Bereich ist.⁴⁹ Generell gilt, dass Individuallösungen dort dominieren, wo neue Markterfordernisse deutlich werden und noch keine erprobten Systeme verfügbar sind oder wo sich eine Unternehmung mit entsprechendem Aufwand ein Alleinstellungsmerkmal im Wettbewerb sichern will. Beispielsweise wird sich in branchenspezifischen Bereichen, wie etwa bei besonderen Fertigungsabläufen, häufig zugunsten einer Individualentwicklung entschieden, um die Einzigartigkeit des

⁴⁵ Vgl. Dorn [2000, S. 201–202]; Stahlknecht u. a. [1999, S. 228–229]; Bernhard u. a. [1999, S. 12]; Kloss [2002, S. 16]

⁴⁶ Vgl. Walther-Klaus [1999]; Picot u. a. [1992, S. 23]; Eikelmann [1995, S. 56–57]

⁴⁷ Vgl. Jochem [1997, S. 21–22].

⁴⁸ Vgl. Barbitsch [1996, S. V].

⁴⁹ Vgl. Jochem [1997, S. 22].

Unternehmens zu behalten und eine Abgrenzung zu den Wettbewerbern zu schaffen.⁵⁰ Sollte dennoch SSW eingesetzt werden, besteht für das jeweilige Unternehmen die Möglichkeit, sich durch Anpassungen der Software und gezielte Zusatzentwicklungen von Konkurrenten abzuheben, wie es auch bei Logistiksystemen üblich ist.⁵¹

Der Web-Händler Amazon ist ein Beispiel aus der Praxis dafür, dass im Logistikbereich nicht immer eine Standardlösung eingesetzt wird, obwohl Möglichkeiten dafür auf dem Markt existieren. Die eingesetzte Logistik-Software bei Amazon ist eine Eigenentwicklung von Amazon.com. Das System wurde den spezifischen Anforderungen des Distributionszentrums angepasst. Amazon sieht die Logistik als ihre Kernkompetenz an, da ein schnellstmöglicher Versand für sie entscheidend ist. Wegen des speziellen Geschäftsmodells, der Sortimentsänderung im 24-Stunden-Takt, gab es kein Logistiksystem auf dem Markt, das eins zu eins hätte übernommen werden können.⁵² Bei der Firma Siemens AG wird z. B. für die Logistik in vielen Unternehmensbereichen und Regionen SAP R/3, also eine SSW, eingesetzt.⁵³

3.2.3 Geschäftsformen

Anhand des E-Commerce, des Handels und des Bankensektors wird die Wirkung der Potenziale und Risiken, die die Auswahl und den Einsatz einer Softwarelösung betreffen, in verschiedenen Wirtschaftsbereichen nachfolgend dargestellt.

Als Erstes wird der Einsatz der Software beim E-Commerce erläutert. Die generelle Meinung ist, dass ein Web-Auftritt mit Standardsoftware günstiger und schneller zu realisieren ist als eine individuelle Entwicklung, welches einem generellen Potenzial der SSW entspricht. Der Grund dafür liegt in dem geringeren Programmieraufwand, denn SSW hat den Vorteil, dass die wichtigsten Funktionen bereits zur Verfügung stehen. Beispielsweise besteht die Möglichkeit, Produkte hierarchisch in Gruppen zu sortieren, damit der Kunde sich schneller zurechtfindet. Weiterhin kann ein Modul für die Kundenregistrierung verwendet werden, damit die Benutzer sich beim ersten Einkauf registrieren lassen können, um nicht bei jedem Einkauf die eigene Lieferanschrift und Kontoverbindung neu angeben zu müssen. Deswegen ist eine Standardsoftwarelösung zu empfehlen, wenn es bei dem Web-Auftritt um typische Shop-Funktionen geht. Wenn die Anforderungen und Bedürfnisse aber deutlich höher sind oder wenn die Grundfunktionen der SSW nicht gebraucht werden, ist eine individuelle Lösung empfehlenswert. Aber auch die Unternehmen, die eine fertige Lösung gewählt haben, können nicht ganz auf individuelle Anpassungen verzichten, spätestens, wenn es darum geht, die Internet-Filiale an die internen Systeme anzuschließen.⁵⁴

⁵⁰ Vgl. Österle [1990, S. 21]; Keil u. a. [1998, S. 851–852]

⁵¹ Vgl. Jochem [1997, S. 22].

⁵² Vgl. Calik [2002]; o. V. [2000, S. 78]

⁵³ Vgl. Keil u. a. [1998, S. 859].

⁵⁴ Vgl. o. V. [1999].

Zwei verschiedene Wege zum Web-Auftritt gehen die Firma Degussa-Hüls Dental, die eine Standardsoftwarelösung von Intershop auswählte und Gate Gourmet Zürich, eine Unternehmung, die Fluggesellschaften mit Mahlzeiten versorgt und zusammen mit seinem Dienstleister eine individuelle Lösung entwarf. Degussa-Hüls Dental entschied sich für eine SSW, weil die Funktionen dieser SSW benötigt wurden, um das Implantatsystem Ankylos im Web zu präsentieren. Außerdem war die Abrechnung in unterschiedlichen Währungen möglich und der Shop konnte in verschiedenen Sprachen ins Netz gebracht werden. Für das Catering-Unternehmen stand fest, dass die Daten von Anfang an aus dem Warenwirtschaftssystem in das Shop-System importiert werden sollten, weshalb es am einfachsten war, das vorhandene SAP R/3 um eine Eigenentwicklung zu ergänzen.⁵⁵

Beim Handel ist die zentrale Datenverwaltung aufgrund der komplexen Datenmengen besonders wichtig, da diese immer aktuell sein müssen. Deshalb gewinnen ERP-Systeme zunehmend an Bedeutung, da diese eine einheitliche Betrachtung und die Konsistenz der Daten über alle Prozessabläufe hinweg sicher stellen. Wichtig ist, dass ein ERP-System die besonderen Anforderungen der Handelsunternehmen abdecken kann, wie z. B. Aktionsabwicklung, Limitplanung, Bestandsbewertung zu Verkaufspreisen etc. Ebenfalls müssen die Veränderungen in der Organisationsstruktur oder in der Ablauforganisation schnell und einfach mit der Software abzubilden sein, beispielsweise bei der Einführung einer neuen Vertriebslinie oder bei der Änderung von Belieferungsverfahren. Da gute Standardlösungen das Potenzial bieten, sich an die Organisationsstruktur und die funktionellen Anforderungen der Unternehmung anpassen zu lassen, ist es möglich, dass dieselbe SSW z. B. bei OBI als auch bei Tchibo oder Kaufhof eingesetzt werden kann, natürlich jeweils in einer individuellen Ausprägung.⁵⁶

Branchen mit relativ wenig Unternehmen, wie z. B. Banken oder auch Versicherungen, werden von Anbietern großer SSW vernachlässigt, da wenige Nachfrager existieren. Bei den Banken gibt es nur eine bekannte Standardlösung: „Kordoba“ von Siemens. Obwohl das Produkt bereits seit vielen Jahren existiert, wird es von Großbanken nicht eingesetzt. Die zentralen Systeme der Banken (z. B. Kontenführung und Zahlungsverkehr) basieren auf ISW.⁵⁷ Obwohl heutzutage in vielen Fällen SSW eingesetzt wird, zeigt dieses Beispiel, dass auch in ISW ein großes Potenzial steckt, sofern sie im richtigen Umfeld verwendet wird.

⁵⁵ Vgl. o. V. [1999].

⁵⁶ Vgl. Competence Site [2002, S. 4–6].

⁵⁷ Vgl. Vaske [2000].

4 Ausnutzung der Potenziale und Vermeidung der Risiken von Standard- und Individualsoftware

4.1 Standardsoftware

Generell gilt, dass die Vorteile nicht gleich Potenziale darstellen, sondern es muss darauf geachtet werden, dass sie richtig von der Unternehmung eingesetzt werden. Bei den Nachteilen sollte bedacht werden, dass die Risiken, die entstehen können, vermieden werden. Wichtig ist auch, dass der Kauf einer qualitativ hochwertigen SSW nicht ausreicht, um ihre Vorteile und Chancen zu realisieren. Die erfolgreiche Einführung durch eine systematische Vorgehensweise spielt ebenfalls eine entscheidende Rolle.⁵⁸ Da dieser Themenkomplex nicht Inhalt dieser Arbeit ist, wird er hier nicht weiter erörtert. Einige Beispiele, wie Vorteile in Potenziale umgewandelt und Risiken vermieden werden können, folgen.

Die Anpassungsfähigkeit der SSW stellt ein Potenzial dar, da Customizing tendenziell schneller durchzuführen ist als eine Neuentwicklung. Es besteht aber die Gefahr, dass dieser Vorteil nur mangelhaft ausgenutzt wird, wodurch eine Individualisierung der Software nicht stattfindet und die Ähnlichkeit zu den Konkurrenten groß bleibt. Dadurch kann ein Verlust an Wettbewerbsvorteilen entstehen. Um dieses Risiko zu vermeiden, sollte die Anpassung der Organisation an die Software bzw. der Software an die Organisation in jedem konkreten Fall gründlich geplant und ausgeführt werden. Die Empfehlungen, ob die Organisation an die SSW angepasst werden soll oder umgekehrt, gehen auseinander.⁵⁹ Pietsch empfiehlt beispielsweise, dass Programmanpassungen eher bei größeren Firmen mit IV-Erfahrung und -Kapazität und mit fest vorgegebener Ablauflogik vorgenommen werden sollten. Organisationsanpassungen sind eher für kleinere und mittlere Unternehmungen mit wenig IV-Erfahrung zu empfehlen, da deren Ablauflogik noch nicht starr vorgegeben ist.⁶⁰ Hansen u. a. empfehlen aber einen Mittelweg zu wählen.⁶¹ Außerdem muss darauf geachtet werden, dass die Kosten und Zeitdauer der Anpassung die der Eigenentwicklung nicht überschreiten.

Obwohl SSW grundsätzlich kostengünstiger als ISW ist, sowie bessere Wartung und Service bietet, besteht das Risiko, vom Hersteller abhängig zu werden. Bei den, mit der SSW-Einführung verbundenen, Prozessänderungen, die häufig vorgenommen werden müssen, sowie bei den Wartungskosten und der Schulung der Mitarbeiter können hohe Ausgaben entstehen, wodurch der Vorteil der geringeren Kosten verloren geht. „Nicht alleine der Verkauf von Standardanwendungssoftware bringt für die Softwarelieferanten das Geld, sondern das nachfolgende ‚People-

⁵⁸ Vgl. Barbitsch [1996, S. 2].

⁵⁹ Vgl. Bernhard u. a. [1999, S. 59]; Abts u. a. [2002, S. 78]; Jäger u. a. [1993, S. 425]

⁶⁰ Vgl. Pietsch [1994, S. 6].

⁶¹ Vgl. Hansen u. a. [1983, S. 69].

Business‘ “. ⁶² Deswegen ist die Gestaltung der Verträge sehr wichtig.

Die Wartungsverträge der Standardsoftwareanbieter sorgen für die Möglichkeit, Updates und Releasewechsel der gelieferten Software zu erhalten, was generell von Vorteil ist. Zu erwähnen wären beispielsweise der Beginn des Jahres 2000 oder die Euro-Einführung: ohne Updates hätte es sicherlich erheblich mehr Probleme mit den Softwarelösungen der Unternehmen gegeben. ⁶³ Dies alles kann aber zum Release-Zwang werden, da unter Umständen die Änderungen erst für die Abläufe der Unternehmung passend konfiguriert werden müssen. Es besteht das Risiko, dass eine Änderung mehr Zeit und Geld kostet, als der daraus folgende Nutzen. ⁶⁴ Probleme können auch auftreten, wenn zwei oder mehr Fremdsysteme miteinander verbunden sind und jedes System einer gesonderten Release-Pflege unterliegt. Es kann für die Unternehmung schwierig sein, die passenden Release-Stände zur richtigen Zeit zu erhalten und mit den vorhandenen Schnittstellen aufeinander abzugleichen. ⁶⁵ Deswegen ist eine sorgfältige Überlegung und Überprüfung bezüglich der Kosten, der Zeit und des Nutzens erforderlich, bevor ein neues Release übernommen wird.

Ein Nachteil der SSW ist, dass es bestimmte organisatorische Abläufe und Regeln vorgibt, deren Änderungen zeit- und kostenaufwändig sind. Aber gerade daraus resultiert das Potenzial, die bestehenden Prozessabläufe zu überprüfen, „festgefahrene“ Strukturen zu überdenken und besser zu gestalten. ⁶⁶

Ein Potenzial der Standardsoftware bieten die Zusatzmodule, die noch auf den Markt gebracht werden können, um die SSW auch für die Firmen attraktiv zu machen, deren Anforderungen momentan den Einsatz von SSW wegen ihrer eingeschränkten Funktionalität nicht erlauben. Große Standardsoftwareanbieter können die zusätzlichen Module sicherlich günstiger herausbringen, als entsprechende Eigenentwicklungen. Auch die Weiterentwicklung der Software bezüglich Markt- und Gesetzanforderungen ist bei SSW kostengünstiger. Wichtig dabei ist, dass zukunftsfähige Softwaresysteme einen hohen Standardisierungsgrad (modularer Aufbau, Verwendung von herstellerübergreifenden IT-Standards), Flexibilität, Benutzerfreundlichkeit, Offenheit und Transparenz haben, damit sie von möglichst vielen unterschiedlichen Unternehmen eingesetzt werden können. ⁶⁷

4.2 Individualsoftware

Der Vorteil der Individualsoftware, die Einzigartigkeit, kann dadurch Potenziale bringen, dass eine Differenzierung gegenüber den Konkurrenten erreicht wird, die im günstigen Fall einen

⁶² Bernhard u. a. [1999, S. 3].

⁶³ Vgl. Schulz u. a. [2000, S. 46].

⁶⁴ Vgl. Schulz u. a. [2000, S. 46]; Bernhard u. a. [1999, S. 3–4]

⁶⁵ Vgl. Arnold [1997, S. 64].

⁶⁶ Vgl. Barbitsch [1996, S. 16–17]; Hansen u. a. [1983, S. 108]

⁶⁷ Vgl. Hagenlocher [2000, S. 62–68].

Wettbewerbsvorteil bedeutet. Auch können besonders innovative Arbeitsabläufe entwickelt werden. Allerdings besteht das potenzielle Risiko, dass heute nur noch wenige Mitarbeiter die Fachkenntnisse haben, um vor Jahren oder sogar Jahrzehnten entwickelte Programmkerne verändern oder nur ersetzen zu können. Somit muss darauf geachtet werden, dass die Abhängigkeit von einzelnen Mitarbeitern vermieden wird. Es besteht sonst das Risiko, dass das notwendige Know-How verloren geht, um das Projekt weiterzuführen, wenn ein wichtiger Programmierer die Firma verlässt. Ein Wissenstransfer auf möglichst mehrere Mitarbeiter sollte stattfinden, um dieses Risiko zu vermeiden.⁶⁸

Weiterhin besteht bei der Individualentwicklung das Risiko, dass keine Erfahrungswerte über die Funktionalität existieren. Der Kunde muss sich ganz auf den Projektplan bzw. den Kostenvorschlag verlassen, wenn es um den wichtigen Faktor der Funktionsfähigkeit geht, aber auch wenn es um die Faktoren Zeit und Geld geht, da ISW-Projekte vielfach nicht im gesetzten Kosten- und Zeitrahmen bleiben. Terminverzug bei der Inbetriebnahme bringt verschiedene Folgeprobleme mit sich.⁶⁹ Ein erfahrener, verlässlicher Systempartner, welcher auch Verfügbarkeit, Service und Unterstützung sicherstellt, kann das Risiko verkleinern.

Ein Vorteil der ISW ist, dass weniger Hardware beansprucht wird. Hier besteht das Sparpotenzial in der Vermeidung zusätzlicher Beschaffungen, da das IT-Umfeld in der Unternehmung bei der Entwicklung der Software berücksichtigt werden kann. Damit die Hardware aber nicht zu einem Hindernis wird, müssen alle „Feinheiten“ schon in der Projektplanungsphase durchdacht werden. Sollten Probleme erst im Echtbetrieb auffallen, ist es bereits zu spät und die Änderungen werden teuer.⁷⁰

Da die Software einem ständigen Wandel unterliegt, bedeutet es, dass die Software nie fertig sein wird. Beispielsweise kann wegen des Technologiewandels bei einer Individualentwicklung das Problem auftreten, dass die Eigenentwicklung auf einem neuen System, z. B. nach dem Austausch des Betriebssystems nicht mehr funktioniert.⁷¹ Im Fall der SSW reagieren die großen Softwareanbieter darauf mit einer neuen Version. Bei der Individuallösung braucht der Kunde eigene Kapazitäten für Entwicklung und Wartung bzw. er muss eine Lösung in Zusammenarbeit mit einem Softwarehaus finden. Um dieses Risiko zu vermeiden, sollte er es Spezialisten überlassen, die Entwicklungen immer wieder nachzuziehen, statt einen Großteil seiner Ressourcen daran zu binden.

Die Benutzerakzeptanz wird ebenfalls als ein Vorteil der ISW genannt. Damit dieser Vorteil wirklich zum Potenzial wird, muss darauf geachtet werden, dass die Benutzer in den Entwicklungsprozess integriert werden, damit sie sich später schneller mit dem System identifizieren

⁶⁸ Vgl. Schulz u. a. [2000, S. 48–49].

⁶⁹ Vgl. Hagenlocher [2000, S. 60–61].

⁷⁰ Vgl. Schulz u. a. [2000, S. 43].

⁷¹ Vgl. Kloss [2002, S. 16–17].

können. Auch zukünftige Anpassungen werden nicht nötig sein, wenn die Anforderungen des Anwenders vor der Entwicklung bekannt sind.

Weiterhin kann es bei vielfachem Einsatz von ISW zu Kompatibilitätsproblemen mit anderen Projekten oder Abteilungen kommen. Wird für viele Vorgänge eine spezielle Software entwickelt, werden unter Umständen Daten an mehreren Stellen gepflegt, was zu einem inkonsistenten und damit nicht mehr nutzbaren Datenbestand führen kann. Auch enge Zusammenarbeit mit anderen Firmen, wie z. B. Zulieferern, wird durch den Einsatz von ISW erschwert, da einheitliche Schnittstellen fehlen. Um dies zu vermeiden, sollten bei einer Individualentwicklung die bestehenden Standards (bzgl. Netzwerke, Schnittstellen zu anderen Standardsoftwaresystemen, Systemsoftware etc.) berücksichtigt werden. Ggf. kann der Entwicklungsaufwand durch den Einsatz vorgefertigter standardisierter Komponenten reduziert werden.⁷²

5 Fazit

Es wurde deutlich, dass eine klare Aussage zum Vorrang von Standard- bzw. Individualsoftware nicht möglich ist, da beide gewisse Potenziale aber auch Risiken mit sich bringen. Deswegen hängt eine Software-Bewertung immer vom speziellen Anforderungsprofil des jeweiligen Anwenders ab und nur aus seiner speziellen Situation wird er ein bestimmtes geeignetes Paket auswählen können. Es kommt für jedes Unternehmen darauf an, zu definieren, was die Software leisten soll. Es gibt nur grundlegende Vorschläge, für welchen Wirtschaftsbereich oder in welcher Abteilung eine Softwareart besser geeignet ist als die andere.

Heute geht der Trend zur Standardsoftware. In mittelständischen Unternehmen gibt es meistens keine ernsthafte Alternative zu Standardsoftware, da die Kapazität und das Know-How für die eigene Entwicklung nicht vorhanden ist. Aber auch größere Betriebe, die seit längerer Zeit eigene Lösungen besitzen, entscheiden sich zunehmend für den Einsatz leistungsfähiger SSW. Während z. B. SAP und Oracle vor allem Konzerne und große mittelständische Unternehmen bedienen, sind die Kunden der kleineren Anbieter von SSW in der Regel kleine und mittlere Unternehmen.⁷³

Das Ziel jeder Unternehmung ist ein möglichst vollkommenes, komplettes, alle Bedürfnisse des Unternehmens abdeckendes System auf dem Markt zu finden. Zwischen Wunsch und Wirklichkeit besteht aber oft eine Lücke. Sind die Anforderungen eines Unternehmens sehr spezifisch, ist eine Individualentwicklung oder aber eine Anpassung der Standardsoftware erforderlich. Wenn das Ziel nicht darin besteht, „Standardunternehmen“ zu schaffen, dann wird es richtig

⁷²Vgl. Schlichtherle [1998, S. 153].

⁷³Vgl. Böhm u. a. [1993, S. 127]; Mertens u. a. [2000, S. 28]; Dorn [2000, S. 201]; Stahlknecht u. a. [1999, S. 303]; Jochem [1997, S. 23–24]

sein, für ein optimales und geeignetes Softwarepaket rechts und links an einem Standardsystem vorbeischaun zu dürfen. Optimal wäre es, die Vorteile beider Softwarearten bestmöglich miteinander zu verbinden und zum Einsatz bringen.

Wichtig ist, dass bei der Auswahl einer geeigneten Softwarelösung nicht nur die Vor- und Nachteile der Softwarearten berücksichtigt werden sollen, sondern dass die damit verbundenen Potenziale und Risiken Aufmerksamkeit finden müssen. Es muss immer darauf geachtet werden, dass die Potenziale ohne richtige Ausnutzung zum Risiko werden können. Ebenso können unnötige Risiken eventuell vermieden werden.

Literaturverzeichnis

Abts, D., Müller, W.: Grundkurs Wirtschaftsinformatik. Eine kompakte und praxisorientierte Einführung. 4. Auflage, Vieweg & Sohn Verlagsgesellschaft mbH, Braunschweig u. a. 2002.

Arnold, B.: Darf's ein bißchen mehr sein?. In: Computerwoche, 43 (1997), S. 63–66.

Barbitsch, C. E.: Einführung integrierter Standardsoftware: Handbuch für eine leistungsfähige Unternehmensorganisation. Carl Hanser Verlag, München u. a. 1996.

Bernhard, M., Heinrich, C.: Softwareauswahlprozesse – Eine strategische Entscheidung – In Kernprozessen weg von monolithischen Anwendungen!. <http://www.ecg-consulting.de/IT-Pflichtenheft-u-SW-Auswahl-Alba-IT-Mgmt-99-06-PDF.pdf>. Erstelldatum: 06.1999, Druckdatum: 29.10.2003.

Böhm, R., Fuchs, E., Pacher, G.: System-Entwicklung in der Wirtschafts-Informatik. 2. Auflage, vdf Verlag der Fachvereine an den schweizerischen Hochschulen und Techniken AG, Zürich 1993.

Calik, K.: Amazonier im Stress. http://www.ecc-handel.de/best_practice/1034241945/. Erstelldatum: 10.2002, Druckdatum: 06.12.2003.

Competence Site: E-Interview „Erfolgreiches IT-Management im Handel“ mit Achim Sieren. [http://www.handel-competence-center.de/handel.nsf/666ECAF1E033443CC1256C2300545B72/\\$File/e-interview_sieren_wincor.pdf](http://www.handel-competence-center.de/handel.nsf/666ECAF1E033443CC1256C2300545B72/$File/e-interview_sieren_wincor.pdf). Erstelldatum: 08.2002, Druckdatum: 26.12.2003.

Diehl, H.-J.: Marketing für betriebswirtschaftliche Standardanwendungssoftware: Bewältigung von Unsicherheit und Spezifität im Systemgeschäft. Gabler, Wiesbaden 2000.

Dorn, J.: Planung von betrieblichen Abläufen durch Standardsoftware – ein Widerspruch?. In: Wirtschaftsinformatik, 42 (2000), S. 201–209.

Eikelmann, H.: Standard- und Individualsoftware richtig mischen. In: Online, 12 (1995), S. 56–58.

Enzler, S.: Methodik der Materialflussorientierten Modellierung von ERP-Systemen. http://www.eco-effizienz.de/eco/download/d_flussmanagement_Modellierung_ERP_Systeme.pdf. Erstelldatum: 07.2003, Druckdatum: 09.12.2003.

Frank, J.: Standard-Software: Kriterien und Methoden zur Beurteilung und Auswahl von Software-Produkten. Verlagsgesellschaft Rudolf Müller GmbH & Co., Köln-Braunsfeld 1977.

Hagenlocher, H.: Modulare Konzepte für Logistik-Software. In: Verein Deutscher Ingenieure (Hrsg.): Risikofaktor Software: Tagung Mannheim, 21./22. November 2000. VDI Verlag GmbH, Düsseldorf 2000, S. 54–74.

- Hansen, H. R., Amsüss, W. L., Frömmer, N. S.: Standardsoftware: Beschaffungspolitik, organisatorische Einsatzbedingungen und Marketing. Springer, Berlin u. a. 1983.
- Jäger, E., Pietsch, M., Mertens, P.: Die Auswahl zwischen alternativen Implementierungen von Geschäftsprozessen in einem Standardsoftwarepaket am Beispiel eines Kfz-Zulieferers. In: Wirtschaftsinformatik, 35 (1993), S. 424–433.
- Jochem, M.: Einführung integrierter Standardsoftware: ein ganzheitlicher Ansatz. Dissertation, Universität – Gesamthochschule – Essen, Essen 1997.
- Keil, C., Lang, C.: Standardsoftware und organisatorische Flexibilität: Eine Untersuchung am Beispiel der Siemens AG. In: zfbf, 50 (1998), S. 847–862.
- Kirchmer, M.: Geschäftsprozessorientierte Einführung von Standardsoftware: Vorgehen zur Realisierung strategischer Ziele. Gabler, Wiesbaden 1996.
- Kloss, K.: Ist Individualsoftware noch zeitgemäß?. In: Informationweek, 19 (2002), S. 16–17.
- Mertens, P. u. a.: Grundzüge der Wirtschaftsinformatik. 6. Auflage, Springer, Berlin u. a. 2000.
- Österle, H.: Unternehmensstrategie und Standardsoftware: Schlüsselentscheidungen für die 90er Jahre. In: Österle, H. (Hrsg.): Integrierte Standardsoftware: Entscheidungshilfen für den Einsatz von Softwarepaketen. Hallbergmoos: AIT, Angewandte Informationstechnik Verlags GmbH, München 1990, S. 11–37.
- o. V.: Web-Tools: E-Commerce: Maßgeschneidert oder von der Stange. <http://www.informationweek.de/print.php3?/channels/channel102/992528b.htm>. Erstelldatum: 25.11.1999, Druckdatum: 04.11.2003.
- o. V.: Special – Logistikstrategie Amazon.de. In: Logistik heute: das aktuelle Fachmagazin für Unternehmen mit Zukunft und offizielles Mitteilungsblatt der Bundesvereinigung Logistik e. V. 3 (2000), S. 78–79.
- Picot, A., Maier, M.: Analyse- und Gestaltungskonzepte für das Outsourcing. In: IM Information Management, 4 (1992), S. 14–27.
- Pietsch, M.: Beiträge zur Konfiguration von Standardsoftware am Beispiel der Geschäftsprozessimplementierung und der Parameterinitialeinstellung bei der Einführung eines großintegrierten PPS-Systems. Dissertation, Friedrich-Alexander-Universität Erlangen-Nürnberg, Nürnberg 1994.
- Remme, M.: Konstruktion von Geschäftsprozessen: Ein modellgestützter Ansatz durch Montage generischer Prozeßpartikel. Gabler, Wiesbaden 1997.
- Schlichtherle, O.: Standardsoftware im Unternehmen erfolgreich einsetzen: eine praxisbezogene Einführung. Verlag Praxiswissen, Dortmund 1998.

Schulz, H., Buck, M.: Vorgehen bei Auswahl und Implementierung von Standardsoftware. In: Verein Deutscher Ingenieure (Hrsg.): Risikofaktor Software: Tagung Mannheim, 21./22. November 2000. VDI Verlag GmbH, Düsseldorf 2000, S. 35–52.

Schwarze, J.: Einführung in die Wirtschaftsinformatik. 3. Auflage, Verlag neue Wirtschafts-Briefe GmbH & Co., Herne u. a. 1994.

Schwarzer, B., Krcmar, H.: Wirtschaftsinformatik: Grundzüge der betrieblichen Datenverarbeitung. Schäffer-Poeschel, Stuttgart 1996.

Stahlknecht, P., Hasenkamp, U.: Einführung in die Wirtschaftsinformatik. 9. Auflage, Springer, Berlin u. a. 1999.

Vaske, H.: Individualsoftware trotz den „Modewellen“ der IT. <http://www.computerwoche.de/index.cfm?pageid=254&artid=13057&type=detail&kw=individualsoftware>. Erstelldatum: 21.02.2000, Druckdatum: 18.12.2003.

Walther-Klaus, E.: Standardsoftware versus Individualsoftware im Mittelstand: Chance oder Risiko?. <http://www.holz-zentralblatt.com/bit.nsf/www/bitart990517>. Erstelldatum: 05.1999, Druckdatum: 25.09.2003.