

September 2001

Standardisierungswettbewerb zwischen parametrisierbarer Standardsoftware und komponentenbasierten Anwendungssystemen

Stefan Volkert

Universität Augsburg, stefan.volkert@wiso.uni-augsburg.de

Bernd Reitwiesner

Universität Augsburg, bernd.reitwiesner@wiso.uni-augsburg.de

Follow this and additional works at: <http://aisel.aisnet.org/wi2001>

Recommended Citation

Volkert, Stefan and Reitwiesner, Bernd, "Standardisierungswettbewerb zwischen parametrisierbarer Standardsoftware und komponentenbasierten Anwendungssystemen" (2001). *Wirtschaftsinformatik Proceedings 2001*. 48.
<http://aisel.aisnet.org/wi2001/48>

This material is brought to you by the Wirtschaftsinformatik at AIS Electronic Library (AISeL). It has been accepted for inclusion in Wirtschaftsinformatik Proceedings 2001 by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

In: Buhl, Hans Ulrich, u.a. (Hg.) 2001. *Information Age Economy*; 5. Internationale Tagung
Wirtschaftsinformatik 2001. Heidelberg: Physica-Verlag

ISBN: 3-7908-1427-X

© Physica-Verlag Heidelberg 2001

Standardisierungswettbewerb zwischen parametrisierbarer Standardsoftware und komponentenbasierten Anwendungssystemen

Stefan Volkert, Bernd Reitwiesner

Universität Augsburg

Zusammenfassung: Aus einer technologischen Perspektive sind Business Frameworks ein vielversprechender Ansatz, Unternehmungen auf effiziente Art und Weise mit betriebswirtschaftlichen Anwendungssystemen zu versorgen. Trotz dieser Vision kann man am Markt bisher nicht feststellen, daß Frameworkanbieter die Marktstellung etablierter Standardsoftwarehersteller bedrohen. In dieser Arbeit wird modellbasiert der Wettbewerb zwischen Herstellern von Standardsoftware und Frameworkanbietern untersucht. Der Fokus liegt dabei auf den zwei wesentlichen Einflußfaktoren: dem Netzeffekt und der Wirkung von Kompatibilitäts- bzw. Standardisierungsentscheidungen. Auf Basis des Modells können Empfehlungen über die strategische Positionierung von Softwarehäusern im Standardisierungswettbewerb abgeleitet werden.

Schlüsselworte: Frameworks, parametrisierbare Standardsoftware, Netzeffekte, Standardisierung, Standardisierungswettbewerb, Markteintrittsabschreckung

1 Einleitung und Grundlagen

Die Entwicklung von Frameworks und Framework-basierter Software wird seit längerer Zeit in der Wirtschaftsinformatik und der Informatik intensiv diskutiert. Frameworks werden dabei vielfach als das Mittel angesehen, die momentan gewaltige Nachfrage nach flexiblen und erweiterbaren Anwendungssystemen für den Electronic Commerce schnell und kostengünstig zu befriedigen. Die auf Basis der Komponententechnologie entwickelten unternehmungsweiten betriebswirtschaftlichen Anwendungssysteme (sog. Enterprise Resource Planning (ERP) – Systeme) stehen dabei in unmittelbarer Konkurrenz zu den auf Basis herkömmlicher parametrisierbarer betriebswirtschaftlicher Standardsoftware entwickelten ERP-Systemen. Zentrales Merkmal Framework-basierter Anwendungssysteme (AwS) ist, daß sie weder völlig eigenentwickelt werden noch vollständig auf Standardkomponenten basieren. Vielmehr werden individuell entwickelte Komponenten mit modifizierten oder unmodifizierten Standardkomponenten auf Basis eines Frameworks zu einem AwS integriert. Dadurch können die Vorteile von Standard- und Individualsoftware miteinander vereinigt werden [MeBo⁺00, S. 147]. Die Zielsetzung ist, Anwendungssysteme zu entwickeln, die genauso exakt die Anforderun-

gen der Anwender erfüllen wie Individualsoftware, aber zu Preisen ähnlich denen von Standardsoftware angeboten werden können.

Trotz dieser augenscheinlichen Vorteile konnten sich bislang Frameworks kaum durchsetzen. Abgesehen von IBM's San Francisco Framework sind es eher kleinere Anbieter wie z. B. Navision, die auf Frameworktechnologie setzen. Noch immer dominiert parametrisierbare Standardsoftware wie SAP's R/3 den Markt für ERP-Systeme. In jüngster Zeit jedoch sind neue Anstrengungen großer Anbieter mit Microsofts .Net- und Oracles .Now-Strategie erkennbar. Aber auch Standardsoftwarehersteller passen sich an ändernde Märkte an: immer häufiger werden Schnittstellen offengelegt, die Drittanbietern die Möglichkeit bieten, mit ihren Produkten die Funktionalität der Anwendungssysteme zu erweitern. So ist es beispielsweise möglich, über die sog. BAPI-Schnittstellen Komponenten anderer Hersteller in das SAP R/3-System zu integrieren. Für die Zukunft ist auf dem Markt für EC-fähige ERP-Systeme weitere Bewegung zu erwarten.

Bei der Analyse des Software-Marktes stellt man fest, dass der Erfolg der Frameworktechnologie viel weniger von rein technischen Einflussfaktoren (siehe [Ivan96],[PhIv97]) als von ökonomischen und marktpolitischen Rahmenbedingungen, wie Marktanteil, Markteintrittsbarrieren, Standardisierung und anderen Faktoren abhängt. Die komponentenbasierten Ansätze können ihre Vorteile erst dann entfalten, wenn es einen hinreichend großen Markt für Softwarekomponenten bzw. eine funktionierende „Softwarekomponenten-Industrie“ gibt. Davon kann derzeit jedoch nicht ausgegangen werden [DiEs201]. In diesem Aufsatz wird deshalb der Wettbewerb zwischen Standardsoftwareherstellern auf der einen und Framework- und Komponentenanbietern auf der anderen Seite aus einer Markt- und Wettbewerbsperspektive betrachtet. Ziel ist es, die ökonomischen Rahmenbedingungen auf dem Markt für ERP-Systeme aufzuzeigen und die entscheidenden Wettbewerbsfaktoren herauszuarbeiten. Es soll dabei insbesondere um die Beantwortung folgender Fragen gehen: Kann die Framework-Technologie den Marktanteil erreichen, der für eine profitable Entwicklung notwendig ist? Lohnen sich demnach die Investitionen in die Framework-Technologie? Soll ein Standardsoftwarehersteller seine Software für Dritthersteller öffnen? Wann kann ein etabliertes proprietäres System als Markteintrittsbarriere dienen? Welche Rolle spielt Standardisierung im Marktprozess? Auf dieser Basis werden dann Gestaltungsempfehlungen für die unterschiedlichen Teilnehmer des Wettbewerbs abgeleitet.

Zunächst soll das dem Aufsatz zugrundeliegende Begriffsverständnis von Frameworks, Framework-basierter Software und Standardsoftware festgelegt werden. *Standardsoftware* ist die Bezeichnung für „Programme, die nicht für einen einzelnen Anwender, sondern für eine Vielzahl von Kunden mit gleichen oder ähnlichen Problemstellungen produziert werden“ [MeBo⁺00, S. 24]. Unter dem Begriff *komponentenbasierte Softwareentwicklung* und *Componentware* werden Softwarekonzepte diskutiert, bei denen Anwendungssysteme weder völlig eigenentwickelt werden noch vollständig auf Standardsoftware basieren. Mit dem Begriff der *Komponente* bezeichnet man „Softwarebausteine, die speziell für die Wiederver-

wendung konzipiert wurden“ [Soef97, S. 82]. Um Komponenten sinnvoll nutzen zu können, ist eine Infrastruktur notwendig, welche steuernde und koordinierende Dienste zur Verfügung stellt und so die einzelnen Komponenten zu einem Gesamtsystem integriert [Soef97, S. 82]. Diese Infrastruktur wird auch als Framework bezeichnet. Im Fortgang der Arbeit werden nur Business Frameworks, also domänenspezifische Frameworks, die die Entwicklung von Geschäftsanwendungen unterstützen [Schm00, S. 2f.] betrachtet.

Abgesehen von den bereits angesprochenen technologischen und methodischen Aspekten der Softwareentwicklung gibt es einige Besonderheiten im Wettbewerb zwischen Frameworks und Standardsoftware, die einen großen Einfluß auf die jeweiligen Marktpositionen dieser konkurrierenden Technologien ausüben. Sie werden verursacht durch den sog. Netzeffekt bzw. Netzwerkeffekt. Hierbei handelt es sich um externe Effekte bei den Konsumenten von Gütern, die dazu führen, daß der Nutzen, welchen ein einzelner Konsument aus dem Konsum eines Gutes zieht, mit der Anzahl der weiteren Nutzer eines bestimmten Gutes ansteigt. Netzeffekte sind ein wichtiger Faktor auf dem Absatzmarkt für Software und können verschiedene Ursachen haben. So wird eine Unternehmung, die plant, die Standardsoftware eines bestimmten Anbieters zur zentralen Plattform für alle Geschäftsprozesse der Unternehmung zu machen, den durch die Software implementierten Prozessen um so mehr vertrauen, je mehr andere Unternehmungen bereits erfolgreich ihre Geschäftsprozesse auf Basis des selben Produkts unterstützen. Zusätzlich zu solchen eher psychologischen Effekten können Netzwerkeffekte auch durch die Notwendigkeit interoperabler Geschäftsprozesse verursacht werden. Möchten beispielsweise zwei oder mehr Unternehmungen im Rahmen eines Wertschöpfungsnetzes zusammenarbeiten, besteht die Notwendigkeit, interorganisationale Geschäftsprozesse auf Basis interoperabler Anwendungssysteme unterstützen und abwickeln zu können. Für ein spezifisches Business Framework wird zudem auch die Zahl der am Markt verfügbaren Software-Komponenten, die problemlos innerhalb dieses Framework genutzt werden können, von der Anzahl der Nutzer des selben Frameworks abhängen.

Falls zwei verschiedene Standardsoftwareprodukte, zwei verschiedene Typen von Framework-Technologien oder eine Standardsoftware und ein Framework bzw. dessen Komponenten zusammenarbeiten können, bezeichnet man sie als *kompatibel*. Es können unterschiedliche Grade bzw. Intensitäten der Kompatibilität unterschieden werden. Ein Produkt kann von den Netzeffekten des anderen Produktes profitieren, falls es teilweise oder vollständig kompatibel zu diesem Produkt ist. Standardisierung, falls der Standard allgemein akzeptiert wird, führt dazu, daß verschiedene Produkte in einem Produktbereich jeweils den gleichen Grad der Kompatibilität zueinander erhalten und in gleichem Maße an den Netzwerkeffekten der Mitbewerber partizipieren können. Nicht zuletzt deshalb haben Kompatibilitäts- und Standardisierungsentscheidungen einen großen Einfluß auf die Wettbewerbssituation auf Software-Märkten.

2 Der Wettbewerb zwischen den Technologien

Fassen wir zunächst nochmals die wettbewerbsrelevanten Eigenschaften der unterschiedlichen Technologien zusammen: ein wesentlicher Unterschied zwischen den Produktmerkmalen der beiden Technologien ist der Grad an Individualität und optimaler Anpassung an die individuellen Anforderungen der nutzenden Unternehmung. Die Anpaßbarkeit parametrisierbarer Standardsoftware an die Anforderungen des Nutzers ist auf die aus der prinzipiell vorhandenen Funktionalität gewinnbaren Flexibilität begrenzt. So ist es letztlich auf Basis parametrisierbarer Standardsoftware nicht möglich, Lösungen zu implementieren, die die Anforderungen der Nutzer vollständig erfüllen und an die Individualität von Individualsoftware heranreichen [Vask00].

Framework-basierte Anwendungssysteme ermöglichen demgegenüber die Implementierung von Lösungen, die auf dem Wege der Anpassung und Erweiterung von Standardbausteinen sowie durch Integration zusätzlicher eigenentwickelter Komponenten genau den unternehmungsspezifischen Anforderungen entsprechen [Schm00, S. 10]. So besteht beispielsweise die typische Anwendung, die auf Basis des IBM San Francisco Framework entwickelt wird, zu 40 % aus dem Framework und den dazugehörigen Komponenten, die nur noch um nutzerspezifische Benutzungsoberflächen, Landes- und Branchenspezifika, Geschäftsregeln und individuelle Anwendungsfunktionen ergänzt werden müssen [Sche98, S. 116].

Trotz der unterschiedlichen Arten der Anpassung von Frameworks und Standardsoftware an die unternehmungsindividuellen Anforderungen muß die Entwicklung und Einführung von Framework-basierter Software nicht notwendigerweise mit höheren Kosten als die Entwicklung und Einführung eines angepaßten Anwendungssystems auf Basis von Standardsoftware verbunden sein. Dies ist darauf zurückzuführen, daß bei Projekten zur unternehmungsweiten Einführung von Standardsoftware das Customizing nur eine von vielen notwendigen Projektaktivitäten darstellt.

Für die Analyse des Wettbewerbs zwischen den beiden Technologien ist eine Abstraktion von den Detailproblemen bei der Entwicklung und Einführung von Anwendungssystemen notwendig. Als für die Analyse der Marktposition der beiden Technologien im Markt für ERP-Systeme wesentlichen Faktoren sind zu berücksichtigen zum Ersten das Ausmaß, in welchem ein Anwendungssystem den Anforderungen des Nutzers genügt. Zum Zweiten die Anzahl der nutzenden Unternehmungen, die bereits eine Installation einer Software eines bestimmten Typs oder Herstellers in Betrieb haben und der Einfluß der Größe dieser *installierten Basis* auf den Nutzen der Unternehmungen aus dieser Software (sog. *Netzeffekt*). Zum Dritten der Einfluß von Kompatibilitäts- und Standardisierungsentscheidungen.

Diese drei Faktoren können mit Hilfe eines Marktmodells abgebildet werden, dessen exakte mathematische Formulierung sich im Anhang befindet (das Modell

baut auf einem in [PFWi98, S. 295ff.], welches um die Einführung asymmetrischer Produktcharakteristika und die Berücksichtigung fixer Kosten modifiziert wird). Darin wird Wettbewerb zwischen zwei Anbietern S und F modelliert, die jeweils Komplettlösungen (Entwicklung und Einführung) unternehmungsweiter ERP-Systeme anbieten. S stellt seine Lösung auf Basis einer parametrisierbaren Standardsoftware bereit, F auf Basis eines Frameworks und den zugehörigen Komponenten. Wir nehmen eine oligopolistische Konkurrenz an, wobei die Anzahl der Anbieter ohne Beschränkung der Allgemeinheit auf zwei reduziert werden kann.

Auf Basis des Modells kann zunächst die Nachfrage nach Anwendungssystemen auf Basis der unterschiedlichen Technologien ermittelt werden (sog. Markennachfragen). Daraus können erste Aussagen über die relativen Wettbewerbspositionen von Standardsoftware und Frameworks abgeleitet werden. Zur Interpretation der Markennachfragen (7) und (8) werden drei Teilterme (a), (b) und (c) unterschieden. Diese lassen sich wie folgt interpretieren:

- “Natürlicher Kundenstamm” (a): Gemäß Annahme (A3) entspricht die Framework-basierte Lösung zu hundert Prozent den Anforderungen des Nutzers. Dies hat zur Folge, daß die Nutzer der Framework-basierten Lösung keine Nutzeneinbußen aus den Produkteigenschaften hinnehmen müssen. Deshalb wird ein potentieller Nutzer sich immer dann, wenn andere Faktoren wie Preis- oder Netznachteile ausgeblendet werden können, für die Framework-basierte Lösung entscheiden. Es ergibt sich für das Framework ein natürlicher Kundenstamm in Höhe von 1, für die Standardsoftware in Höhe von 0.
- Preisvorteil (b): Es sei zu Erklärungszwecken angenommen, daß die Standardsoftware-basierte Lösung zu einem geringeren Preis zu haben sei als die Framework-basierte Lösung. Je größer dieser Preisvorteil der Standardsoftware ausfällt, um so mehr Nachfrager werden sich für die Standardsoftware-basierte Lösung entscheiden und damit deren Marktanteil erhöhen.
- Netzvorteil (c): Dieser Term beschreibt den Einfluß des für Softwaremärkte wichtigen Netzeffekts. Es sei zu Erklärungszwecken ebenfalls angenommen, daß die Standardsoftware schon wesentlich länger am Markt etabliert sei als das Framework und deshalb von einer hohen installierten Basis aus früheren Perioden profitieren kann, wohingegen das Framework erst in der letzten Periode in den Markt eingetreten ist und dementsprechend nur über eine geringe installierte Basis verfügt. Je größer dieser Unterschied der installierten Basen zwischen den Technologien ausfällt, um so größer wird der Marktanteil der Standardsoftware auch in der aktuellen Periode sein. Zu Bedenken ist jedoch, daß F seinen Netznachteil durch die Wahl eines hohen Kompatibilitätsgrades s_F reduzieren kann. Denn die Kompatibilitätsgrade s_S und s_F wirken auf den Netzvorteil Δn_i von S (vgl. Formel (10)) über den Zusammenhang $(s_S x_F^i - s_F x_S^i)$. Während Δn_i den Netzvorteil, der aus dem Absatz vergangener Perioden resultiert, repräsentiert, beschreibt der verbleibende Term $(s_S - 1)$ den

Einfluß der Absatzunterschiede zwischen den beiden Produkten in der aktuellen Periode. Deshalb ergibt sich insgesamt nur dann ein Netzvorteil für S , falls der Netzvorteil aus vergangenen Perioden die Gesamtnachfrage (normiert auf 1) nach Software aus dieser Periode, verringert um den Kompatibilitätsgrad der Standardsoftware, übersteigt:

$$\Delta n^i > 1 - s_s$$

Ergebnisse des Preiswettbewerbs bei gegebener Kompatibilität

Ist der Kompatibilitätsgrad der Produkte der beiden Anbieter (als Ergebnis des Standardisierungswettbewerbs (vgl. Abschnitt 3) fixiert, bleibt den Anbietern als taktische Variable der Preis, zu dem die Produkte angeboten werden. Da ein simultaner duopolistischer Wettbewerb angenommen wurde (A6) ist es für den einzelnen Wettbewerber notwendig, bei der Festlegung des eigenen Preises die Preispolitik des Konkurrenten mit zu berücksichtigen. Bei der Analyse der Gewinne im Marktgleichgewicht stellen sich insbesondere die folgenden beiden Fragen:

- (a) Werden Gewinne größer Null erzielt?
- (b) Welcher Anbieter erzielt den höheren Gewinn?

Es kann gezeigt werden, daß die Preise im Gleichgewicht immer dann größer sind als die marginalen Kosten c , so lange der Marktanteil eines Wettbewerbers größer als Null ist. Dies ist aufgrund der Annahme 6 des duopolistischen Wettbewerbs immer der Fall. Beide Wettbewerber können demnach immer einen positiven Deckungsbeitrag erzielen und erzielen dann einen Gewinn, falls die fixen Kosten FC klein genug sind. Da die Gesamtnachfrage völlig unelastisch ist (A2), werden die Nachfrager zwar c.p. beim günstigeren Anbieter kaufen. Dennoch besteht für keinen der beiden Anbieter ein Anreiz, in diesem einperiodigen Spiel einen nicht fixkostendeckenden Preis zu verlangen. Beide Anbieter können im Gleichgewicht demnach die Fixkosten decken und Extragewinne erzielen.

Die Antwort auf Teilfrage (b) hängt wesentlich von der Ausprägung der jeweiligen Einflußfaktoren ab. Die Bedingung dafür, daß der Gewinn von F (π_F^B) den Gewinn (π_S^B) von S übersteigt, läßt sich wie folgt formalisieren:

$$\pi_F^B > \pi_S^B \Leftrightarrow \frac{1}{6\lambda} - \frac{2}{3}e(\Delta n^i + s_s - 1) > 0 \quad (1)$$

Der Gewinn des Framework-Anbieters übersteigt den Gewinn des Standardsoftware-Anbieters im Falle einer geringen Wettbewerbsintensität und eines geringen Netzvorteils der Standardsoftware gegenüber dem Framework (dies kann beispielsweise durch die Wahl eines hohen Kompatibilitätsgrades s_F erreicht werden). Geht man beispielsweise davon aus, daß der Standardsoftware-Hersteller über eine weit größere installierte Basis verfügt als der Framework-Hersteller, dann ist es

für den Framework-Hersteller nötig, hinreichend kompatibel zu S produzieren zu können, um einen höheren Gewinn zu erzielen als dieser.

Es sich lohnt demnach unter folgenden Bedingungen für den prospektiven Hersteller der Framework-basierten Lösung, das mit der Entwicklung des Frameworks verbundene Investitionsrisiko einzugehen: gelingt es der Framework-Technologie und damit dem Hersteller Framework-basierter Lösungen, einen relevanten Marktanteil im Markt für betriebswirtschaftliche Anwendungssysteme zu erhalten, dann können die zur Entwicklung der Framework-Technologie notwendigen Aufwendungen kompensiert werden. Zudem können unter den diskutierten Bedingungen über die in den fixen und variablen Kosten bereits enthaltene „normale“ Gewinnspanne hinaus Extragewinne erzielt werden.

Insbesondere die Wahl des Kompatibilitätsgrades hat einen bedeutenden Einfluß auf die Gewinne im Marktgleichgewicht. Im nächsten Abschnitt sollen deshalb die wesentlichen Faktoren im Kompatibilitätswettbewerb der ersten Stufe, dem sogenannten Standardisierungswettbewerb, aufgezeigt werden.

3 Standardisierungswettbewerb

Aufgrund der hohen Bedeutung der Netzeffekte im Softwaremarkt bringt hohe Kompatibilität zum Konkurrenzprodukt zumindest für den Anbieter mit der geringeren installierten Basis und dem geringeren erwarteten Marktanteil in der aktuellen Periode erhebliche Vorteile mit sich. Allerdings muß dabei bedacht werden, daß ein gewinnmaximierender Kompatibilitätsgrad auf Softwaremärkten üblicherweise nicht ohne Mitwirkung und ggf. Zustimmung des Konkurrenten, zu dem man kompatibel sein möchte, erreicht werden kann. Standardisierung, mit deren Hilfe Kompatibilität zwischen Produkten unterschiedlicher Hersteller erreicht werden kann, spielt demnach eine große Rolle. Ziel eines Standards im Hinblick auf Software ist es üblicherweise, die Interoperabilität zwischen Softwarekomponenten, die die Anforderungen des Standards erfüllen, zu garantieren. Ein von beiden Wettbewerbern akzeptierter Schnittstellenstandard würde in dem hier präsentierten Modell beiden Herstellern identische Kompatibilitätsgrade, aber nicht notwendigerweise vollständige Kompatibilität ermöglichen. Im Modell führt die Einführung eines Standards zu einem einheitlichen Kompatibilitätsgrad für beide Anbieter. Die Aussagen des vorigen Abschnitts 2 lassen sich auch auf den Fall der Standardisierung, d.h. der Existenz eines akzeptierten Standards, übertragen.

Zudem kann auf Basis des Modells die Frage nach dem gewinnoptimalen Standardisierungsgrad untersucht werden. So ist es denkbar, daß Konkurrenten auf Softwaremärkten grundsätzlich bereit sind am Standardisierungsprozeß mitzuwirken, jedoch unterschiedliche Zielsetzungen im Hinblick auf die Höhe des durch

den Standard erreichbaren Kompatibilitätsgrades verfolgen. Für die Frage nach dem wirtschaftlichen Interesse an einer Standardisierung bzw. Ausweitung der gegenwärtigen Standardisierung soll der Gewinnverlauf sowohl für den Standardsoftwarehersteller als auch für den Hersteller Framework-basierter Software in Abhängigkeit der installierten Basis der Standardsoftware untersucht werden.

Dazu sei zunächst der Gewinn von S bei variablem Standardisierungsgrad s in Abhängigkeit von der installierten Basis an Standardsoftware untersucht. Abbildung 1 zeigt die Abhängigkeit zwischen $\partial\pi_s^B/\partial s$ und x_S^i . Der Graph ist eine nach unten geöffnete Parabel mit 2 Nullstellen. Die rechte Nullstelle (x_2) ist immer im positiven Bereich von x_S^i , während sich für die linke Nullstelle (x_1) keine allgemeine Aussage treffen lässt. Die Abbildung zeigt, dass es einen Bereich ($x_S^i > x_2$) gibt, indem eine Erhöhung des Standardisierungsgrades den Gewinn des Standardsoftwareanbieters vermindert. Die Analyse jedoch auch, daß unterhalb des kritischen Wertes x_2 die zunehmende Standardisierung auch aus Sicht des Standardsoftwareanbieters vorteilhaft ist. Man erkennt also, dass der Standardsoftwarehersteller nicht in jedem Fall, sondern erst ab einer hinreichend großen installierten Basis fortschreitenden Standardisierungsbemühungen ablehnend gegenüber stehen sollte.

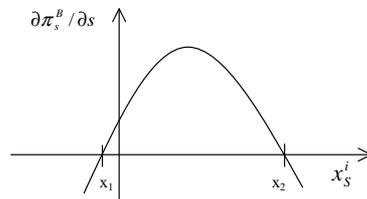


Abbildung 1

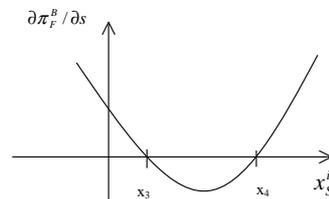


Abbildung 2

Die Gewinnanalyse für F bei variablem Standardisierungsgrad s in Abhängigkeit der installierten Basis wird in Abbildung 2 veranschaulicht. Diese zeigt den typischen Zusammenhang zwischen $\partial\pi_F^B/\partial s$ und x_S^i . Der Graph ist eine nach oben geöffnete Parabel mit 2 Nullstellen. Der interessante Bereich liegt rechts von x_4 : bei einer installierten Basis von $x_S^i > x_4$ profitiert ein Anbieter Framework-basierter Software von einer weiteren Standardisierung, er ist so in der Lage, sich den Netzeffekt der installierten Basis der Standardsoftware zueigen zu machen.

4 Einschränkungen und Ausblick

Das hier präsentierte Modell wurde so einfach wie möglich gehalten, um dessen formale Lösbarkeit und die Interpretierbarkeit der Ergebnisse sicherzustellen. Dies führt zu einigen Einschränkungen. Die offensichtlichsten der Einschränkungen sowie Möglichkeiten zu deren Abhilfe sollen kurz besprochen werden. So wurden in Annahme 5 wurden marginale Kosten bzw. identische Stückkosten sowie identische Fixkosten für S und F angenommen und begründet. Es können aber auch unterschiedliche Kosten in das Modell aufgenommen sowie eine Lösung dafür angegeben werden, auch wenn es dadurch schwierig wird, allgemeingültig eindeutige Aussagen zu treffen. Zudem liegt dem Modell liegt eine verhältnismäßig einfache Definition von Kompatibilität zu Grunde. Diese ist jedoch ausreichend zur Diskussion der Beziehungen zwischen Lösungen auf Basis von Standardsoftware und komponentenbasierter Software als Ganzes, insbesondere vor dem Hintergrund, daß derzeit noch keine Märkte für den Zukauf fachlicher Komponenten existieren. Darüber hinaus beinhaltet das Modell beinhaltet keine zeitliche Komponente. Eine Erweiterung um zusätzliche Parameter wäre jedoch möglich.

5 Implikationen der Analyse

Auf dem Markt für ERP-Systeme konkurrieren die komponentenbasierten Ansätze mit den etablierten Produkten der Anbieter parametrisierbarer Standardsoftware. Dabei stellen sich für die Anbieter Framework-basierter Lösungen und die Anbieter parametrisierbarer Standardsoftware unterschiedliche Herausforderungen, die aus den unterschiedlichen Produkteigenschaften sowie den verschiedenen Marktpositionen der beiden Technologien resultieren. In diesem Beitrag wurde deshalb die Wettbewerbssituation bei der Bereitstellung von unternehmensweiten Anwendungssystemen auf Basis von Standardsoftware vs. auf Basis von Frameworks mit Hilfe eines mikroökonomischen Wettbewerbsmodells analysiert.

Auf Basis des Modells konnte gezeigt werden, daß es im Falle eines positiv verlaufenden Markteintritts für den Anbieter der Framework-basierten Lösung möglich ist, den Aufwand für die Produktentwicklung zu decken. Darüber hinaus können unter den in Abschnitt 2 aufgezeigten Bedingungen Extragewinne erzielt werden. Insofern ist es lohnend, das Entwicklungsrisiko einzugehen, falls ein Markteintritt möglich ist. Die Netzeffekte der etablierten Hersteller stellen hier das entscheidende Markteintrittshemmnis dar. Ein allgemein akzeptierter und umgesetzter Standard für die Interaktion von Softwarekomponenten kann dazu beitragen, daß die Hersteller Framework-basierter Software an den Netzeffekten der etablierten Software partizipieren können und so die Markteintrittshemmnisse verringert werden. Es kann jedoch auch im Standardisierungsfall der Markteintritt verhindert werden, falls im Standardisierungsprozeß nur ein geringer Standardisie-

rungsgrad realisiert werden kann, welcher den Limit-Standardisierungsgrad nicht überschreitet. Die etablierten Hersteller parametrisierbarer Standardsoftware sollten demnach nicht den Standardisierungsprozeß blockieren, sondern gezielt am Standardisierungsprozeß mitwirken, um den für sie besten Standardisierungsgrad durchsetzen zu können.

Allerdings besteht auch für die Anbieter traditioneller betriebswirtschaftlicher Standardsoftware Handlungsbedarf. Deren Hauptkunden - große Industrieunternehmen - sind mittlerweile weitgehend mit betriebswirtschaftlicher Standardsoftware versorgt, wodurch der Markt enger wird. So stellt sich auch für diese Anbieter die Frage, wie sie ihren Marktanteil verbessern können. Hier sind vor allem zwei strategische Optionen zu sehen: zum einen die Verbesserung der Anpassungsmöglichkeiten an die individuellen Anforderungen des Nutzers und zum anderen eine Zerlegung des Gesamtsystems in Komponenten, welche dann auch einzeln verkauft werden können. Die Auswirkungen dieser beiden strategischen Optionen auf die Marktposition der Hersteller betriebswirtschaftlicher Standardsoftware können ebenfalls anhand der Modellanalyse und deren Ergebnisse diskutiert werden.

Die erste der beiden Optionen setzt an dem im Modell abgebildeten zentralen Nachteil der parametrisierbaren Standardsoftware an: deren im Verhältnis zu Framework-basierten Lösungen vergleichsweise geringere Anpaßbarkeit an die individuellen Anforderungen des Nutzers, welche dieser mit einem Nutzenabschlag bewertet. Hier wurden im Modell zwei Extrempositionen angenommen: völlige Anpaßbarkeit für die Framework-basierte Lösung, keine Anpaßbarkeit für die parametrisierbare Software. In der Realität dürften diese Extrempositionen so zwar nicht gegeben sein, dennoch ist der Grad der Anpaßbarkeit an die Anforderungen und damit der Grad der Erfüllung der Anforderungen für Standardsoftware als wesentlich geringer einzustufen als für die Framework-basierten Lösungen. Im Modell kann dieser Nachteil ausgeglichen werden durch die Einräumung von Preisvorteilen oder die Ausnutzung von Netzvorteilen gegenüber dem Mitbewerber. In der Realität besteht eine weitere Option für den Hersteller parametrisierbarer Software darin, den Grad der Anpaßbarkeit an die individuellen Anforderungen durch eine Änderung des Produktes zu verbessern. So könnten Parametrisierungsmöglichkeiten weiter ausgefeilt und generischer gestaltet werden. Will man jedoch bei der Parametrisierung als zentrale Anpassungsmöglichkeit bleiben, da dieses Anpassungsparadigma eine zentrale Voraussetzung für ein reibungsloses Upgrade auf neuere Versionen einer Standardsoftware darstellt (vgl. [BrHe⁺01, S. 7]), dann sind dieser Anpaßbarkeit jedoch Grenzen gesetzt (vgl. [KeLa98, S. 856]). So kann immer nur die konkrete Ausprägung der Ausführung der vorher festgelegten Funktionalität durch Parameter gesteuert, nicht jedoch die Funktionalität selbst variiert werden.

Die andere strategische Option, die bereits wahrgenommen wird (so plant die SAP AG, in Zukunft Anwendungskomponenten von mySAP.com auch einzeln zu verkaufen, vgl. [oV01]), besteht in der Zerlegung der Gesamtsoftware in einzeln

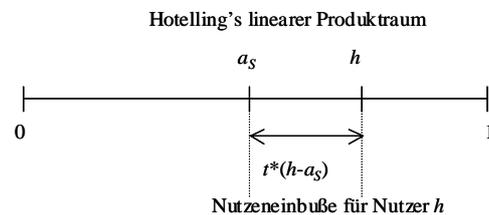
nutz- und verkaufbare Komponenten. Dadurch kommen auch kleine und mittlere Unternehmungen als potentielle Kunden in Frage, für die das Gesamtpaket möglicherweise zu teuer wäre. Sollen diese aus Gesamtlösungen extrahierten Komponenten auch mit Anwendungssystemen anderer Hersteller zusammenarbeiten können, ist jedoch ebenfalls die Anlehnung an einen Standard für die Interaktion von Komponenten erforderlich. Dies kann aber bedeuten, daß Vorteile aus dem Netzeffekt aus der hohen installierten Basis aufgegeben werden müssen. Wird die Strategie der Komponentisierung existierender Systeme eingeschlagen, dann ist demnach kritisch abzuwägen, bis zu welchem Standardisierungsgrad die Umsetzung des Standards sinnvoll erscheint. In Abschnitt 3 wurde hierzu aufgezeigt, daß bei nur geringem Basisvorteil des Standardsoftwareherstellers ein Kooperation am Standardisierungsprozeß sinnvoll erscheint. Ist der Basisvorteil des Standardsoftwareherstellers sehr groß, dann sollte der Limit-Standardisierungsgrad jedoch nicht überschritten werden. Für den Fall der vollständigen Standardisierung kann auf Basis des Modells gezeigt werden, daß unter den hier getroffenen Annahmen der Gewinn des Anbieters der Framework-basierten Lösung den Gewinn des Anbieters der Standardsoftware-basierten Lösung immer übersteigen wird. Je höher der Standardisierungsgrad gewählt wird, desto mehr wird der Standardsoftwarehersteller demnach daran arbeiten müssen, den Nachteil der geringeren Anpaßbarkeit an die Nutzerspezifischen Anforderungen auszugleichen. Ist dies nur in geringem Ausmaß möglich, dann ist die Verhinderung der Standardisierung mit dem Ziel des Erhalts der Markteintrittsbarrieren die bessere Strategie.

Ob die Strategie der Komponentisierung bestehender Systeme für die etablierten Anbieter von Standardsoftware den richtigen Weg darstellt, wird in der Realität jedoch von weiteren Faktoren abhängen. So führt beispielsweise der Electronic Commerce mit der steigenden Anzahl der automatisierten Abwicklung von Transaktionen zwischen Unternehmungen auf Basis von elektronischen Marktplätzen zur verstärkten Forderung nach der Standardisierung von Geschäftsprozessen auch von Unternehmungen außerhalb der Softwareindustrie. Ist abzusehen, daß der Markteintritt von Anbietern Framework-basierter Lösungen nicht verhindert werden kann, dann sollte die Strategie der Komponentisierung von den Anbietern etablierter Systeme jedoch nachhaltig verfolgt werden. Aus Sicht der Nachfrager nach Software wäre dies zu begrüßen, da dies zu einer qualitativ besseren Softwareversorgung der Unternehmungen führen würde.

Anhang: Formulierung des Modells/Annahmen

Die Modell beruht auf einem Ansatz von Hotelling [Hote29] zur Modellierung von Produkteigenschaften heterogener Produkte.

- (A1) Die Position eines einzelnen Nutzers im Produktraum wird mit dem Buchstaben $h \in (0,1]$ bezeichnet. Die Nachfrager sind in diesem gleichverteilt.
- (A2) Die Gesamtnachfrage nach Software-Lösungen ist fix und auf 1 normiert.
- (A3) Die auf Basis der Framework-Technologie von F entwickelte Lösung erfüllt zu 100% die Anforderungen eines jeden Nutzers entlang der horizontalen Linie. Bei Standardsoftware führen Abweichungen der Produkteigenschaften a_S von den Anforderungen h zu Nutzeneinbußen d :



$$d_S(h) = t \cdot |h - a_S| \text{ für die Nutzer von } S \quad (2)$$

$$d_F(h) = 0 \quad \forall h \in (0,1] \text{ für die Nutzer von } F \quad (3)$$

t ist als Heterogenitätsparameter ein Maß, wie wichtig es einem Nutzer ist, eine Softwarelösung zu erhalten, die seinen Anforderungen entspricht.

- (A4) S wählt die Position $a_S = 0,5$ in der Mitte des Produktraumes.
- (A5) Beide Lösungsanbieter können ihre Lösung zu identischen marginalen Kosten bzw. (variablen) Stückkosten pro Einheit Software c erstellen. Diese werden als konstant angenommen.
- (A6) Es herrscht duopolistischer Wettbewerb.
- (A7) Beide Anbieter verhalten sich gewinnmaximierend. Es wird ein einperiodiger Wettbewerb auf Basis vollständiger Information angenommen. Die Gewinne π_l errechnen sich für Preise p_l und Mengen x_l wie folgt:

$$\pi_l = (p_l - c) \cdot x_l - FC, \quad l \in \{S, F\} \quad (4)$$

- (A8) Die Nutzer entscheiden sich für das Produkt mit der höheren Konsumentenrente CS_l (mit $l \in \{S, F\}$). Diese besteht aus dem Bruttonutzen/der Bruttozahlungsbereitschaft Z , vermindert um den effektiven Preis der Softwarelösung p_l sowie die Nutzeneinbuße, erhöht um den Netznutzen en_l :

$$CS_F = Z - p_F - t \cdot 0 + en_F \quad (5)$$

$$CS_S = Z - p_S - t \cdot |h - a_S| + en_S \quad (6)$$

Die Netzeffektstärke $e > 0$ stellt ein Maß für die generelle Wertschätzung der Netzeffekte durch die Nutzer dar. n_l ist das Maß für die Netzgröße.

- (A9) Die Nutzer der Software haben vollständige Information über die Preise und die Kompatibilitätsgrade beider Technologien.
- (A10) Der Wettbewerb zwischen F und S findet in zwei Stufen jeweils simultan statt: auf der ersten Stufe (*Kompatibilitätswettbewerb*) entscheiden die Wettbewerber über den Kompatibilitätsgrad s_F und s_S ihrer Produkte. Auf der zweiten Stufe (*Preiswettbewerb*) werden dann die Kompatibilitätsgrade und damit die Netzwerkvorteile/-nachteile als gegeben angenommen. Der *Kompatibilitätsgrad* s_l gibt dabei den Anteil an Funktionalität an, den Software l von Software m nutzen kann
- (A11) Der indifferente Konsument präferiert die Framework-basierte Lösung.

Nach Auflösung des absoluten Betrags $|h - a_S|$ (mit Fallunterscheidung ($h \geq a_S$ vs. $h < a_S$)) ergeben sich unter der Annahme rationaler Erwartungen der Nachfrager über den Marktanteil der verschiedenen Technologien folgende sog. Markennachfragen:

$$x_F = 1 - 2\lambda[(p_F - p_S) + e(\Delta n^i + s_S - 1)] \quad (7)$$

$$x_S = 0 + 2\lambda \underbrace{(p_F - p_S)}_{(a)} + \underbrace{e(\Delta n^i + s_S - 1)}_{(c)} \quad (8)$$

$$\lambda = \frac{1}{t - 2e(2 - s_S - s_F)} \text{ Wettbewerbsintensität} \quad (9)$$

$$\Delta n^i = (x_S^i - x_F^i) + (s_S x_F^i - s_F x_S^i) \text{ Netzeffektvorteil von } S \quad (10)$$

Literatur

- [BrHe⁺01] Brehm, L./Heinzl, A./Markus, M.L.: Tailoring ERP Systems: A Spectrum of Choices and their Implications. In: Proceedings of the Hawai'i International Conference on System Sciences, January 3-6, Maui, Hawaii.

- [DiEs01] Dietzsch, A./Esswein, W.: Gibt es eine „Softwarekomponenten Industrie ? Ergebnisse einer empirischen Untersuchung. In: Buhl, H.U./Huther, A./Reitwiesner, B.: Information Age Economy, Heidelberg 2001.
- [Hote29] Hotelling, H. [1929]: Stability in Competition. In: Economic Journal 39, S. 41 - 57.
- [Ivan96] Ivanov, E. [1996]: Entwicklung und Anwendung von Frameworks – Probleme und Lösungsansätze, 1996. (<http://www.theoinf.tu-ilmenau.de/~ivanov/scripts/studie.ps>, Download am 12/2/1999).
- [KeLa98] Keil, C./Lang, C. [1998]: Standardsoftware und organisatorische Flexibilität – Eine Untersuchung am Beispiel der Siemens AG. In: ZfbF 50 (1998) 9, S. 847 – 862.
- [MeBo⁺00] Mertens, P./Bodendorf, F./König, W./Picot, A./Schumann, M.: Grundzüge der Wirtschaftsinformatik, 6. Auflage, 2000.
- [oV01] SAP verkauft Mysap auch in Modulen. In: Computerwoche, 27. Jahrgang (2001) Nr. 5, S. 1.
- [PhIv97] Philippow, I./Ivanov, E. [1997]: Methodische Entwicklung von Frameworks. In: 42. Internationales Wissenschaftliches Kolloquium der TU Ilmenau 22.-25.9. 1997, Band 1, S. 15 - 23.
- [PFWi98] Pfähler, W./Wiese, H. [1998]: Unternehmensstrategien im Wettbewerb, Eine spieltheoretische Analyse.
- [Sche98] Scheer, A.W. [1998]: ARIS - Vom Geschäftsprozeß zum Anwendungssystem.
- [Schm00] Schmitzer, B.: Klassifikationsaspekte betriebswirtschaftlich orientierter Frameworks, Forwin-Bericht-Nr.: FWN-2000-006, Bayerischer Forschungsverbund Wirtschaftsinformatik, 2000.
- [Soef97] Soeffky, M.: Componentware. In: Mertens, P. (Hrsg.), Lexikon der Wirtschaftsinformatik, 3. Auflage, 1997, S. 82 – 83.
- [Vask00] Vaske, H. [2000]: Individualsoftware trotz den “Modewellen” in der IT (Interview mit E. Denert). In: Computerwoche 8/2000, S.30 - 31.