

COMPUTER ASSISTED LEARNING

CAL + CAT

COMPUTER ASSISTED TEACHING

ARBEITSBERICHTE

Nr. 28

**Heinz Lothar Grob
Frank Bensberg
Blasius Lofi Dewanto
Maik A. Lindner**

**Ein Open Source-basiertes
Learning Managementsystem
für Volksbanken**

November 2004

**PROF. DR. HEINZ LOTHAR GROB
INSTITUT FÜR WIRTSCHAFTSINFORMATIK
WESTFÄLISCHE WILHELMS-UNIVERSITÄT MÜNSTER**

LEONARDO-CAMPUS 3, 48149 MÜNSTER, TEL. (0251) 83-38000, FAX (0251) 83-38009
E-MAIL: Grob@wi.uni-muenster.de, WWW: <http://www-wi.uni-muenster.de/aw>

Inhalt

1	Motivation	1
2	Rechtliche, ökonomische und technologische Grundlagen von Open Source-Software	5
3	Gestaltung eines Open Source-Learning Managementsystems für Volksbanken	7
3.1	Konzeptionelle Grundlagen der Systemgestaltung	7
3.2	Systemarchitektur	9
3.3	Funktionen	11
3.4	Organisation des Entwicklungsprozesses	14
3.5	Wirtschaftlichkeitsbetrachtung	16
4	Fazit	17
	Literatur	18

1 Motivation

Wettbewerbsdruck und ein zunehmender Wissenswandel führen dazu, dass Unternehmen kontinuierlichen Innovations- und Veränderungsprozessen unterliegen und zunehmend steigende Anforderungen an die Mitarbeiterqualifizierung gestellt werden.¹ Von diesem branchenübergreifenden Phänomen sind auch Genossenschaftsbanken betroffen, die angesichts des Strukturwandels der Kreditwirtschaft einem erhöhtem Konsolidierungsdruck in sämtlichen Geschäftsbereichen ausgesetzt sind.² Leistungsfähige Konzepte zur Personalentwicklung stellen daher einen wesentlichen Strategiebaustein zur nachhaltigen Stärkung der Wettbewerbsposition des Genossenschaftlichen Bankenverbands dar. Ein Erfolg versprechendes Instrumentarium zur Unterstützung dieser personalwirtschaftlichen Aufgabenstellung stellt das *E-Learning* bereit. Hierunter sind wirtschaftsinformationssystemische Konzepte zur Gestaltung und Nutzung von Informations- und Kommunikationssystemen (IKS) zur Abwicklung von Lehr- und Lernprozessen zu verstehen.

Zur Unterstützung von Lernprozessen werden in der betrieblichen und universitären Praxis *Learning Managementsysteme* (LMS) eingesetzt, die auf der Grundlage des Internets Funktionen zur Distribution von Lehr- und Lernmaterialien und zur Kommunikation zwischen Lehrenden und Lernenden anbieten.³ Mit dem Einsatz derartiger E-Learning-Plattformen gehen unterschiedliche Zielsetzungen einher.⁴ Während *Effizienzziele* die nachhaltige Optimierung von Arbeitsabläufen durch die Realisierung von Rationalisierungsnutzeffekten beinhalten, verfolgen *Innovationsziele* die Gestaltung neuartiger Leistungen, die mithilfe von multimedialen Informations- und Kommunikationstechnologien umgesetzt werden können, z. B. personalisierte Lernformen. Allerdings ist festzustellen, dass hinsichtlich der Definition des LMS-Begriffs keine Einheitlichkeit herrscht. So versteht die *Learning Technology Systems Architecture* (LTSA) der IEEE unter einem LMS eine Komponente, die Mechanismen zur Begleitung, Motivation und Einflussnahme auf den Lernfortschritt zur Verfügung stellt.⁵ In Beiträgen, die die komparative Analyse unterschiedlicher LMS-Produkte zum Gegenstand haben, ist jedoch häufig eine breitere Begriffsauffassung anzutreffen,⁶ sodass auch hier unter einem LMS ein *administratives Informations- und Kommunikationssystem* (IKS) verstanden wird, das Lernende und Lehrende bei der Verwaltung, Distribution, Kommunikation und Organisation von Lerninhalten unterstützt.

¹ Vgl. Hettrich, A., Koroleva, N. (2003), S. 1-3.

² Vgl. Eim, A. (2004), S. 12-13; Lindemann, J. (2004), S. 9-10.

³ Vgl. Grob, H. L., Bensberg, F., Dewanto, B. L., Düppe, I. (2004), S. 46 ff.; Grob, H. L., Bensberg, F., Dewanto, B. L. (2004), S. 387-393.

⁴ Vgl. Doberkat, E., Veltmann, C., Engels, G., Hausmann, J. H., Lohmann, M. (2002), S. 2-7.

⁵ Vgl. IEEE (2001), S. 76, S. 85.

⁶ Vgl. Baumgartner, P., Häfele, H., Maier-Häfele, K. (2002).

Die Analyse des LMS-Markts zeigt, dass eine Vielzahl von Softwareprodukten existiert, die mittlerweile ein hohes Maß an funktionaler Stabilität aufweisen.¹ In Bezug auf die softwarebezogenen Merkmale der *Herkunft* und *Distributionsform* der verfügbaren Produkte lassen sich zwei LMS-Softwarearten voneinander abgrenzen. Zum einen stehen *proprietäre* LMS-Produkte zur Verfügung, die von den Herstellern unter engen lizenzrechtlichen Bestimmungen distribuiert werden. Zum anderen haben die intensiven Entwicklungsaktivitäten der Hochschulen dazu geführt, dass mittlerweile eine Vielzahl *freier* LMS-Softwareprodukte vorhanden ist, die nach den Lizenzbestimmungen der *Open Source* angeboten werden. Als Beispiel für die gestiegene Marktrelevanz von Open Source-LMS ist das Produktportfolio des Mediatoren CampusSource anzuführen, das mittlerweile mehr als 16 E-Learning-Plattformen umfasst.² Die zentralen Merkmale dieser diametralen LMS-Softwarearten werden in der folgenden Abbildung gegenübergestellt.

Proprietäre LMS (Closed Source)	Freie LMS (Open Source)
Keine Verfügbarkeit des Quelltextes für den Anwender	Publikation des vollständigen Quelltextes in verständlicher Form
Modifikation, Vervielfältigung und Verbreitung des Softwareprodukts werden lizenzvertraglich unterbunden	Modifikation, Vervielfältigung und Verbreitung werden lizenzvertraglich gestattet
Erwerb des Nutzungsrechts erfolgt gegen Entgelt	Nutzungsrecht wird ohne Zahlung eines Entgelts erworben

Abb. 1: Unterschiede zwischen proprietären und freien LMS³

Dabei resultiert aus dem Open Source-Ansatz zur Gestaltung und Nutzung von E-Learning-Plattformen eine Reihe von strategischen Erfolgsfaktoren, die nicht nur für öffentliche Bildungseinrichtungen, sondern vielmehr auch für Unternehmungen von hohem Interesse sind. Diese Erfolgsfaktoren können in Bezug auf die *informationstechnologische*, *organisatorische* und *ökonomische* Wirkungsebene differenziert werden (vgl. Abb. 2).

¹ Eine umfassende LMS-Marktstudie liefern Hettrich, A., Koroleva, N. (2003).

² Die Initiative *CampusSource* wurde vom Ministerium für Schule, Wissenschaft und Forschung des Landes Nordrhein-Westfalen zum Aufbau eines Virtuellen Hochschulraums NRW initiiert. Das Ziel dieser Open Source-Initiative ist es, informationstechnologische Plattformen zur *Nutzung und Weiterentwicklung* auf der Grundlage der Lizenzbedingungen von Open Source zur Verfügung zu stellen. Vgl. Köster (2001); Grob, H. L., Bensberg, F., Kaderali, F. (2004 a), S. 233-242; Grob, H. L., Bensberg, F., Kaderali, F. (2004 b), S. 504-508.

³ Vgl. Grob, H. L., Bensberg, F., Reepmeyer, J.-A. (2004), S. 278.

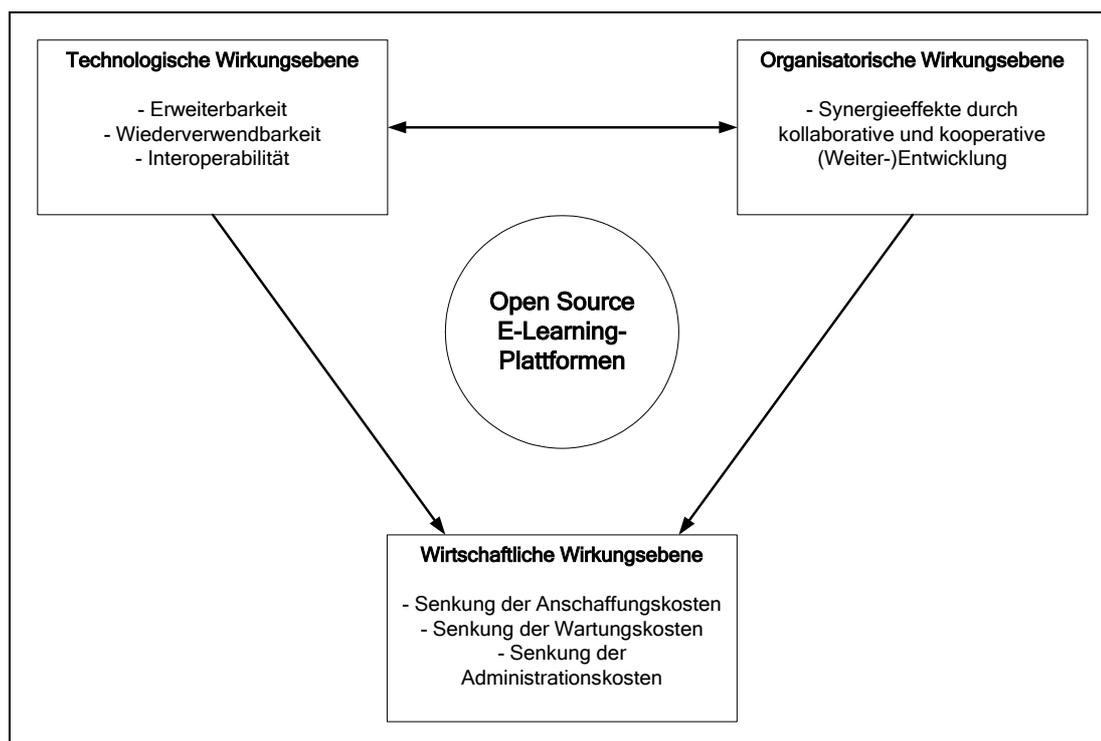


Abb. 2: Potenziale von Open Source-E-Learning-Plattformen¹

Der *technologische* Erfolgsfaktor ist die hohe Flexibilität von Open Source-basierten Softwareprodukten, die insbesondere durch die Eigenschaften der *Erweiterbarkeit*, *Wiederverwendbarkeit* und *Interoperabilität* begründet ist. Diese softwaretechnischen Eigenschaften stellen sicher, dass Open Source-Produkte dynamisch an wandelnde Anforderungsszenarien und Umweltsituationen angepasst werden können. In Bezug auf die *Organisation* des Produktionsprozesses von Open Source-Software (OSS) ist festzustellen, dass dieser keiner hierarchischen Koordinationsform unterliegt, sondern regelmäßig durch selbst organisierte Netzwerke erfolgt. Diese Organisationsform weist signifikante Parallelen zur genossenschaftlichen Organisationsform auf² und führt zu *Synergieeffekten*, die in einer hohen Entwicklungsdynamik durch emergierende dezentrale Softwareinnovationen zum Ausdruck kommen. Aus diesen technologischen und organisatorischen Erfolgsfaktoren resultieren auch positive Effekte bezüglich der *Wirtschaftlichkeit* von OSS, die sich insbesondere durch geringere lebenszyklusbezogene Gesamtkosten auszeichnen, die Gegenstand dedizierter TCO-Analysen sind.³

¹ In enger Anlehnung an Grob, H. L., Bensberg, F. (2003), S. 14.

² Vgl. Rahmen-Zurek, K. (2000), S. 32.

³ Eine Erfassung der monetären Konsequenzen von *proprietären* vs. *freien* LMS liefern Buchholz, H., Leybold, M. E., Schilling, T. (2003). Zur finanzwirtschaftlichen Bewertung von LMS-Produkten mithilfe des TCO-Ansatzes vgl. Bensberg, F., Dewanto, B. L. (2003), S. 9-12. Zur Fundierung einer finanzwirtschaftlich begründeten TCO-Methodik vgl. Grob, H. L., Lahme, N. (2004).

Die skizzierten Eigenschaften von Open Source-LMS werfen insbesondere für den hier thematisierten Aufgabenbereich der informationstechnologischen Unterstützung der Aus- und Weiterbildung in Genossenschaftsbanken interessante Forschungsfragen auf. Die Parallelen zwischen genossenschaftlichen Organisationsformen und der Organisation von Open Source-Softwareprojekten führen zur Hypothese, dass derartige Softwaresysteme für Genossenschaften ein höheres Nutzenpotenzial aufweisen als proprietäre Softwareprodukte.¹ Im Aufgabenkontext des E-Learnings können Nutzeffekte aus der inhärenten Flexibilität von Open Source-Softwareprodukten resultieren. So stellen derartige Softwareprodukte im Rahmen eines *Resource Based Views* immaterielle Ressourcen dar, deren Anwendung nicht nur Rationalisierungsnutzen durch geringere TCO eröffnen, sondern auch zur Gestaltung innovativer Unternehmensstrategien und Geschäftsmodelle beitragen können.² Voraussetzung hierfür ist, dass Unternehmen derartige Softwareressourcen in die eigenen Forschungs- und Entwicklungsprozesse integrieren und Kernkompetenzen zur strategischen Adaption aufbauen. Unter diesen Voraussetzungen können Open Source-LMS zur Generierung *distinktiver Ressourcen* führen, deren geschäftstypenspezifische Anwendung eine strategische Prämisse zur Erzielung und Sicherung von Wettbewerbsvorteilen darstellt.³

Damit Open Source-LMS ihre Nutzenpotenziale im genossenschaftlichen Organisationskontext entfalten können, sind domänenspezifische Adaptionleistungen erforderlich. Bei der funktionalen und inhaltlichen Analyse verfügbarer Open Source-LMS ist festzustellen, dass sämtliche marktseitig gegebenen Systeme für die Domäne der Hochschullehre entwickelt wurden, sodass eine Systemadoption durch Genossenschaftsbanken nicht *uno actu* möglich ist.⁴ Vielmehr sind Modifikationen bezüglich der zu unterstützenden *Lernprozesse* (Anwendungsschicht) und der *Benutzeroberfläche* (Präsentationsschicht) notwendig. Dieser Sachverhalt motiviert das Forschungsprojekt *LMS für Volksbanken* am Institut für Wirtschaftsinformatik, das die domänenspezifische Adaption des Open Source-LMS OpenUSS für Genossenschaftsbanken zum Gegenstand hat und somit zum Abbau des ressourcenbezogenen Anpassungsbedarfs beiträgt. Im Rahmen dieses Projekts wird eine E-Learning-Plattform konzipiert, die anforderungsgerechte Dienste für die Anwendungsdomäne der Volksbanken und Raiffeisenbanken zur Verfügung stellt und aufgrund ihrer inhärenten Flexibilität zur nachhaltigen Steigerung der genossenschaftlichen Kernkompetenzen beiträgt.

¹ Zur Evaluation des IT-Einsatzes in *selbst organisierten Gemeinschaften* vgl. Naumann, S., Schmitz, A. (2004).

² Zur Abstimmung von betrieblicher Anwendungsarchitektur und Unternehmensstrategie vgl. Grob, H. L., Bensberg, F., Reepmeyer, J.-A. (2004), S. 400.

³ Zum *Resource Based View* im Kontext des Genossenschaftlichen Bankenverbands vgl. Eim, A. (2004), S. 36. Eine Reflexion der Resource Based View-Theorie im Marktkontext von Open Source-Betriebssystemen vollzieht Müller-Lietzkow, J. (2004).

⁴ Vgl. hierzu das Produktportfolio des Open Source-Mediatoren *CampusSource* im Internet unter <http://www.campussource.de>.

Im Folgenden werden die Entwicklungsergebnisse des Forschungsprojekts *LMS für Volksbanken* dargestellt. Zur Einordnung des Open Source-Entwicklungsansatzes erfolgt zunächst die Darstellung der rechtlichen, technischen und ökonomischen Grundlagen von Open Source-Software. Hierauf aufbauend werden die Architektur und die Funktionalität der entwickelten Lernplattform *OpenUSS für Volksbanken* vorgestellt.

2 Rechtliche, ökonomische und technologische Grundlagen von Open Source-Software

Mit der zunehmenden wirtschaftlichen Bedeutung branchenunabhängiger Systemsoftware, wie z. B. Linux oder Apache, hat das Phänomen Open Source die Softwareindustrie nachhaltig beeinflusst. Neben der wachsenden Zahl von Softwareunternehmen, die ihre Entwicklungsaktivitäten nach Open Source-Prinzipien strukturieren¹, folgen in zunehmendem Maße auch Bildungseinrichtungen bei der Einführung und Entwicklung von Softwaresystemen dem Paradigma offener Software. Dabei ist offene Software kein Phänomen der 90er-Jahre, sondern lässt sich bis zu den Wurzeln des Betriebssystems Unix zurückverfolgen, das in den 70er-Jahren entwickelt wurde. Aufgrund kartellrechtlicher Bestimmungen war es AT&T zunächst untersagt, Unix kommerziell zu verwerten. Die Lizenzierung der Unix-Quellen gegen nominelle Lizenzgebühren führte schließlich zu einer intensiven Verbreitung und Weiterentwicklung im akademischen Bereich. Als im Jahr 1984 AT&T das Recht zugesprochen wurde, im Softwaremarkt aktiv zu werden, kam es zu einer deutlichen Anhebung der Lizenzgebühren. Als Gegenreaktion gründete Richard Stallman, einer der Pioniere der freien Softwarebewegung, das Projekt GNU (ein rekursives Akronym für „GNU is not Unix“) und entwickelte die GNU General Public License (GPL)², die für die heutige Open Source-Bewegung von maßgeblicher Bedeutung ist. Die GPL gestattet zunächst die Erstellung und Verbreitung unveränderter Kopien des Programmquellcodes, sofern bestimmte Bedingungen (z. B. Vermerk des Copyrights, Ausschluss der Haftung) erfüllt sind. Darüber hinaus regelt die GPL auch die Weiterentwicklung und anschließende Verbreitung modifizierter Programmversionen. Diese dürfen vervielfältigt und verbreitet werden, sofern die vorgenommenen Änderungen kenntlich gemacht werden und die Verbreitung der Bearbeitung ebenfalls im Rahmen der GPL erfolgt.³ Eine Kombination von quelloffener Software mit proprietärer Software, die unter restriktiven Lizenzbedingungen steht, ist gemäß GPL nicht gestattet. Zur Wahrung wirtschaftlicher Interessen gibt es zudem eine Vielzahl herstellerepezifischer Lizenzmodelle, die dem Lizenzinhaber besondere Rechte einräumen.⁴ Um für Entwickler und Anwender Klarheit in Bezug auf die rechtlichen Bestimmungen zu schaffen, vergibt die Open Source-Initiative (OSI) ein eige-

¹ Vgl. Friedewald, M., Blind, K., Edler, J. (2002), S. 158 f.

² Vgl. Ljungberg, J. (2000), S. 209.

³ Vgl. Pearson, H. (2000), S. 153 f.

⁴ Vgl. Bensberg, F., Dewanto, L. (2001).

nes Prüfsiegel (OSI certified) für Softwareprodukte, welche die notwendigen rechtlichen Qualitätsmerkmale quelloffener Software aufweisen und der Open Source-Definition entsprechen. Demnach sind an Open Source-Produkte insbesondere folgende Anforderungen zu stellen:¹

- Veröffentlichung des vollständigen Quellcodes des Softwareprodukts in verständlicher Form und
- Gestattung der Nutzung des Softwareprodukts durch den Urheber, einschließlich technischer Veränderungen sowie die Weitergabe an Dritte.

Im ökonomischen Beziehungszusammenhang ist die Begriffsbildung der „freien“ Software differenziert zu interpretieren. Bisweilen wird hiermit eine *kostenfreie* Bereitstellung des Softwareprodukts vorausgesetzt. Zwar trifft dies für eine Vielzahl von Open Source-Produkten zu, die z. B. per Internet zum Download zur Verfügung gestellt werden, doch dürfen für die Bereitstellung der Software durchaus Gebühren erhoben werden. Dabei handelt es sich allerdings nicht um Lizenzgebühren für die Nutzung der Software, sondern um die Distributionskosten des Anbieters.² Darüber hinaus ist zu berücksichtigen, dass der Einsatz freier Software weitere Kosten verursacht. Wie auch bei proprietärer Software (i. d. R. Closed Source) entsteht ein Großteil der softwarebezogenen Gesamtkosten durch die Installation, die Anwenderschulung und den Support. Für diese Leistungen haben sich softwarezentrierte Geschäftsmodelle entwickelt, die ein umfassendes Betätigungsfeld für wirtschaftliche Aktivitäten in diesem Softwaresektor bieten. Das derzeit wohl geläufigste Geschäftsmodell ist dabei das Angebot marktfähiger Dienstleistungen (*Support Selling*). Neben Distribution, Schulung und Beratung werden häufig auch Weiterentwicklungen und Portierungen nach Kundenwunsch durchgeführt (z. B. im Rahmen der Linux-Distributionen SuSe und Red Hat).

Auf der organisatorischen Ebene erfolgt die Entwicklung von Open Source-Software häufig dezentralisiert auf der Basis kooperativer Strukturen. Während klassische Softwareentwicklungsprojekte im Rahmen hierarchischer Organisationsstrukturen durchgeführt werden, besitzen Open Source-Entwicklergemeinschaften häufig den Charakter dynamischer Netzwerke.³ Infolgedessen besitzen die einzelnen Mitentwickler häufig keine fest definierten Aufgabenbereiche, sondern wählen ihren Aktivitätsbereich im Rahmen des Projekts nach der subjektiven Interessenlage (Selbstselektion). Da die Beteiligung an einem Projekt nicht weisungsgebunden erfolgt, setzt der Open Source-Entwicklungsprozess ein hohes Maß an Dialogorientierung voraus und benötigt funktionspezifische Instrumente zur Koordination der – im Extremfall – global verteilten Entwicklergemeinde. Die Kommunikation erfolgt dabei über Basisdienste des Internets (z. B. Diskussionsforen, E-Mail-Verteiler, FAQ-Listen). Darüber hinaus werden bei mittleren bis größeren Projekten auch entwicklungsspezifische Mehrwertdienste benötigt,

¹ Vgl. Sandl, U. (2001), S. 346 f.

² Vgl. Lutterbeck, B., Gehring, R., Horns, A. (2000), S. 74.

³ Vgl. Rahmen-Zurek, K. (2000), S. 24.

wie z. B. die Publikation und Versionierung des Quellcodes in einer zentralen Datenbank oder das systematische Dokumentieren und Nachverfolgen von Fehlern. Zur Unterstützung einer verteilten, kollaborativen Softwareentwicklung steht im Internet eine Vielzahl unterschiedlicher Werkzeuge und Entwicklungsplattformen zur freien Verfügung, die auch die Distribution von Open Source-Softwareprodukten unterstützen. Die offene Gestaltung des Entwicklungsprozesses führt dazu, dass Open Source-Produkte spezifische technologische Eigenschaften aufweisen:

- Durch die Transparenz des Entwicklungsprozesses weisen Open Source-Softwareprodukte tendenziell ein höheres Sicherheitsniveau als proprietäre Entwicklungen auf. So können Entwickler Open Source-Software auf Sicherheitslücken überprüfen und diese im Rahmen der Entwicklergemeinde kommunizieren.¹
- Um herstellereigentliche Systembindungen (Lock in-Effekte) zu vermeiden, verwenden Open Source-Produkte in der Regel offene Standards, die eine Interoperabilität mit bereits existierenden Softwarelandschaften herstellen und den Wartungsaufwand reduzieren. Somit können Anwendungen realisiert werden, die relativ geringe systemspezifische Investitionen erfordern und infolgedessen geringe Nutzungsbarrieren aufweisen.
- Open Source-Entwicklungen basieren häufig auf einer modularen Architektur, die auf eine Wiederverwendung und Erweiterung des Quellcodes ausgerichtet ist.² Auf diese Weise wird sichergestellt, dass das Softwareprodukt gemäß den Anforderungen der Nutzer evolutionär weiterentwickelt und auch als Inputfaktor für Sekundärentwicklungen verwendet werden kann. Hierdurch ist – im Vergleich zur Anwendung proprietärer Standardsoftware – eine höhere Gestaltungsfreiheit gegeben, die zur Differenzierung vom Wettbewerb genutzt werden kann.

Diese technologischen Eigenschaften belegen, dass Open Source-basierte Softwareprodukte über einen hohen Flexibilitätsgrad verfügen und somit einen ressourcenorientierten Beitrag liefern können, um innovative Marktleistungen und Geschäftsmodelle zu generieren. Voraussetzung hierfür ist jedoch eine domänenspezifische Anpassung. Im Folgenden wird dies für die Domäne der *Genossenschaftsbanken* anhand des Open Source-LMS *OpenUSS* gezeigt.

3 Gestaltung eines Open Source-Learning Managementsystems für Volksbanken

3.1 Konzeptionelle Grundlagen der Systemgestaltung

Zu den Kernaufgaben der Personalentwicklung gehört die zielorientierte Aus- und Weiterbildung der Mitarbeiter. Zur Ergänzung und Erweiterung traditioneller Instrumente der Perso-

¹ Vgl. Sandl, U. (2001), S. 349.

² Vgl. O'Reilly, T. (1999), S. 37.

Personalentwicklung (z. B. Präsenzseminare) werden verstärkt innovative Lernformen eingesetzt. Die Vermittlung von Fakten- und Routinewissen erfolgt in zunehmendem Maße durch interaktive E-Learning-Anwendungen. Auf diese Weise wird ein individuelles, selbstorganisiertes und mobiles Lernen ermöglicht. Dieses Instrumentarium kann auch im Rahmen von Blended Learning-Ansätzen mit traditionellen Personalentwicklungsinstrumenten, wie etwa Präsenzseminaren, verknüpft werden, sodass Synergieeffekte erzielt werden. Auch die Kommunikation zwischen den Mitarbeitern im Lernprozess gewinnt zunehmend an Bedeutung. Durch die Implementierung von internetbasierten Kommunikationsinstrumenten kann die Wissensvermittlung innerhalb von Lern- und Arbeitsgruppen (*family groups*) positiv stimuliert werden.

Zur Unterstützung dieser Personalentwicklungsmaßnahmen mithilfe eines Open Source-LMS wird das Softwareprodukt OpenUSS zugrunde gelegt. Dieses Open Source-LMS, dessen Entwicklung an der Westfälischen Wilhelms-Universität Münster initiiert wurde, wird mittlerweile von zahlreichen Universitäten und insgesamt mehr als 20.000 Nutzern eingesetzt. Die weite Verbreitung im internationalen Verbund hat dazu geführt, dass im Rahmen eines kooperativen Entwicklungsansatzes zahlreiche Funktionalitäten entwickelt und in ein komponentenorientiertes Architekturmodell integriert wurden. Diese Architektur gestattet die Zerlegung des Gesamtsystems in weitgehend homogene Teilmodule, die dezentral gepflegt und ergänzt werden. Aufgrund der hohen Modularität und der damit verbundenen Flexibilität erweist sich dieses System für das hier thematisierte Entwicklungsprojekt langfristig als besonders Erfolg versprechend. Die Software bietet zur Vermeidung von Prozessverlusten konsistente personalisierte Sichten für Lernende und Lehrende. Insgesamt werden durch das System die in der folgenden Abbildung dargestellten Funktionalitäten zum Management von *Personalentwicklungsmaßnahmen* zur Verfügung gestellt.

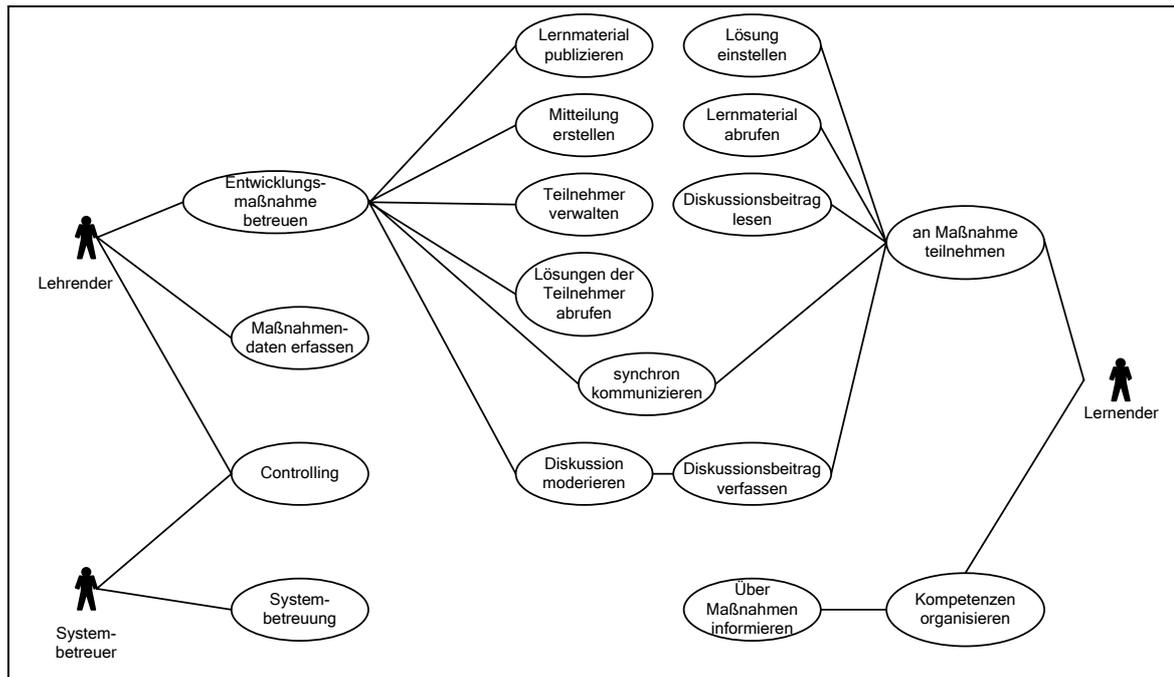


Abb. 3: Funktionsangebot von *OpenUSS für Volksbanken* zum Management von Personalentwicklungsmaßnahmen

Im Folgenden wird zunächst die Architektur des Systems *OpenUSS für Volksbanken* beschrieben. Hierauf aufbauend werden die E-Learning-Dienste skizziert, die vom System bereitgestellt werden.

3.2 Systemarchitektur

Die Infrastruktur von OpenUSS besteht aus Basiskomponenten (Foundation Components), die Dienste für Erweiterungskomponenten (Extension bzw. Pluggable Components) anbieten. Die Sammlung von Basiskomponenten beinhaltet z. B. die Studierenden-, Dozenten-, Administrator-, Organisationseinheits-, Semester-, Themen- und Registrierungskomponente. Darauf aufbauend verfügt OpenUSS über eine Reihe von Anwendungskomponenten, die anwendungsorientierte Funktionalitäten des OpenUSS-Systems enthalten. Sie können einzeln per „Plug & Play“ auf den Basiskomponenten installiert werden (vgl. Abb. 4).

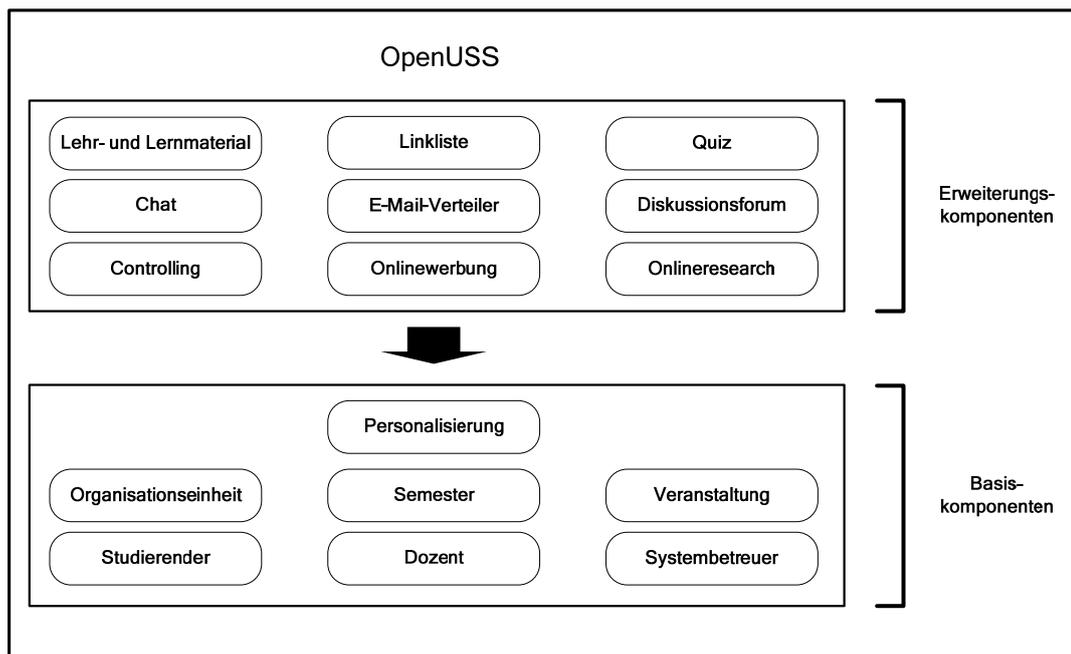


Abb. 4: OpenUSS-Architektur

Das dargestellte Architekturkonzept ermöglicht zum einen eine einfache Erweiterung der *Basis-komponenten*. Durch die klar definierten Schnittstellen der Basiskomponenten können neue Erweiterungskomponenten effizient entworfen und implementiert werden. In Anwendungsdomänen, wie der Aus- und Weiterbildung in der Bankwirtschaft, die durch dynamische Anforderungsszenarien gekennzeichnet ist, eröffnet dieses Architekturkonzept einen signifikanten zukunftsbezogenen Handlungsspielraum, der eine evolutionäre Weiterentwicklung des Gesamtsystems gestattet. Zum anderen ist in dem gewählten Architekturmodell die Wiederverwendung einzelner Module vorgesehen. So lassen sich die – weitgehend domänenunabhängigen – *Erweiterungskomponenten* durch Anpassung der Schnittstellen weiterverwenden und können für weitere Open Source-Softwareprojekte genutzt werden, z. B. die Funktionalitäten der Chat-Komponente, die eine synchrone Kommunikation zwischen Dozenten und Lernenden ermöglicht. Von diesem Ansatz gehen somit auch positive Gestaltungsbeiträge für Softwareentwicklungsprojekte in anderen Anwendungsdomänen aus. Bei der Systementwicklung ist allerdings zu beachten, dass die Entwicklung wieder verwendbarer Komponenten einen deutlich höheren Aufwand erfordert als die einer Einwegversion.¹

Das dargestellte Komponentenmodell wurde mit Technologien implementiert, die den Anforderungen der computergestützten Personalentwicklung entsprechen. Institutionen verfügen meist über heterogen gewachsene Anwendungslandschaften, die sich durch eine Vielzahl unterschiedlicher Hard- und Softwarearchitekturen auszeichnen. Um ein Höchstmaß an Kompatibilität mit bereits existierenden Plattformen herzustellen, wurde als Implementierungstechnologie Java 2 Enterprise Edition (J2EE) gewählt. Diese Softwarearchitektur gestattet eine

¹ Vgl. Rost, J. (1997), S. 361 f.

hohe Unabhängigkeit von existierenden Hardware- und Betriebssystemplattformen und bietet durch die Nutzung offener Standards ein großes Maß an Interoperabilität mit bereits bestehenden Anwendungsarchitekturen. Dies wird insbesondere in Bezug auf komplementäre Softwareprodukte deutlich, die zum Betrieb von OpenUSS notwendig sind. So benötigt OpenUSS zur Speicherung sämtlicher maßnahmenbezogenen Daten zwar ein relationales Datenbanksystem, ist dabei aber nicht auf ein konkretes Datenbankprodukt festgelegt. So können zur Datenspeicherung bereits vorhandene kommerzielle Datenbanksysteme (z. B. Oracle) oder Open Source-Produkte (z. B. InterBase, PostgreSQL) zum Einsatz kommen. Ebenso besteht Unabhängigkeit in Bezug auf die Betriebssystemplattform (z. B. MS Windows, Linux). Auf diese Weise werden potenzielle Anwendungsbarrieren durch inkompatible Systemplattformen abgebaut und eine flexible Skalierung an unterschiedliche Leistungsanforderungen ermöglicht.

3.3 Funktionen

Der Systemzugang erfolgt über einen WWW-Browser über das OpenUSS-Portal (vgl. Abb. 3). Bei der ersten Kontaktaufnahme hat sich der Anwender – also der Dozierende bzw. der Lernende – zu registrieren. Während für den Dozenten vorab interne organisatorische Prozesse zur Vergabe von Rechten abzulaufen haben, kann der Lernende im Normalfall ohne Restriktionen Teilnehmer des Systems werden.

Sponsored by Wirtschaftsinformatik and Controlling

Volksbanken Raiffeisenbanken

Home | Benutzer-Handbücher | Anwenderforum | Entwicklerplattform

Sprache
Sprache
Ändern

Willkommen bei OpenUSS! In Ihrem Browser müssen die Cookies aktiviert sein. Eingabemasken zur Registrierung und zum Login finden Sie auf der linken Seite!

Student in
Benutzername
Passwort
login
...hier registrieren!
Haben Sie Ihr Passwort vergessen?

Dozent in
Benutzername
Passwort
login
...hier registrieren!
Haben Sie Ihr Passwort vergessen?
Neue Institution?
...hier registrieren!

Organisation
[Institutionen finden](#)
[Informationen über diesen OpenUSS-Anbieter](#)

OpenUSS-Barometer
Institutionen: 1
Dozenten: 1
Studenten: 3

WholsOnline?
Wir haben 0 Benutzer online.

Nachrichten von Institutionen
Akademie Deutscher Genossenschaften - 09.11.04: Willkommen in ADG! [mehr](#)
Akademie Deutscher Genossenschaften - 09.11.04: Semesteranfang 2005 [mehr](#)

(c) Copyright OpenUSS 2000 - 2004 devanto@uni-muenster.de
Letzte Änderung: 09.11.04

Abb. 5: Das OpenUSS-Portal

Die zur Registrierung und anschließenden Personalisierung des Teilnehmerzugangs erforderlichen Daten werden dialogorientiert erhoben. Die folgende Abbildung zeigt die entsprechende Bildschirmmaske, die der Datenerhebung dient. Zu beachten ist hierbei, dass die Teilnehmerdaten *personenbezogen* erhoben werden. Infolgedessen sind bei dem Betrieb von OpenUSS zunächst die rechtlichen Bestimmungen des Bundesdatenschutzgesetzes (BDSG) sowie des Teledienststedatenschutzgesetzes (TDDSG) zu berücksichtigen.¹

Sponsored by Wirtschaftsinformatik and Controlling
Volksbanken Raiffeisenbanken

Home | Benutzer-Handbücher | Anwenderforum | Entwicklerplattform

Registrierung für Studierende

2. Schritt: Bitte füllen Sie das Formular mit Ihren Daten aus

Vorname	Conny
Nachname	Controlling
Benutzername*	conny
Passwort	*****
Passwort wiederholen	*****
E-Mail-Adresse	conny@web.de
Address	Controlling Weg 20
City	Münster
Postcode>	48151
Country	Deutschland
Telephone number	0251-1341556
Jahr (optional)	1970
Studiengang (optional)	Master-Studium
Matrikelnummer (optional)	141451
Sprachauswahl	Deutsch

* Eine Unterscheidung in groß/klein-Schreibweise findet nicht statt. Falls Ihr Benutzername bereits vergeben ist, wählen Sie bitte einen anderen.
Sie werden eine Bestätigungs-E-Mail bekommen. Vielen Dank für Ihre Registrierung und haben Sie viel Spaß mit OpenUSS!

Abb. 6: Angabe der persönlichen Nutzerdaten

Bei der Erhebung der persönlichen Nutzerdaten sind insbesondere der Benutzername und das Passwort hervorzuheben. Diese erschließen die Möglichkeit, die Benutzeroberfläche entsprechend der inhaltlichen und formalen Präferenzen des Anwenders zu modellieren und somit einen hohen Personalisierungsgrad zu realisieren. Auf diese Weise wird die informationstechnologische Grundlage zur Errichtung einer *adaptiven Lernplattform* geschaffen, die durch eine präferenzorientierte Gestaltung der Präsentationsschicht ein hohes Effizienzniveau der abzuwickelnden Lernprozesse sicherstellt. Voraussetzung hierfür ist, dass sich der Nutzer vor dem Interaktionsprozess mit der Lernplattform über seine Benutzerkennung und sein Passwort anmeldet (*Login*).

¹ Einen Überblick der informationstechnologischen und organisatorischen Implikationen dieser Rechtsbestimmungen liefern Grob, H. L., Reepmeyer, J.-A., Bensberg, F. (2004), S. 453-461.

The screenshot displays the user interface of the OpenUSS system. At the top, it is sponsored by 'Wirtschaftsinformatik and Controlling' and 'Volksbanken Raiffeisenbanken'. A navigation bar includes links for 'Mein Arbeitsplatz', 'Abo', 'Persönliche Daten', 'Anwenderforum', and 'Logout'. The main content area is personalized for the user 'Home'. A welcome message from 'Conny Controlling' is displayed. Below this, instructions are provided for navigating the system. A 'WholsOnline?' section indicates that one user is online. The page features several interactive elements: 'Abonnierte Veranstaltungen', 'Ausgewählte Fächer', and 'Ausgewählte Institutionen'. A search box allows users to find events by institution or ID. There is also a section for 'Nachrichten von Institutionen' with recent news items. The footer contains copyright information for OpenUSS 2000-2004 and the last update date of 09.11.04.

Abb. 7: Individuelle Startseite

Neben dieser rollenbezogenen Personalisierung verfügt das System über weitere Mechanismen, die eine Individualisierung der Lernsichten erlauben. So hat der Anwender die Möglichkeit, das gewünschte Informationsangebot aufgrund seiner persönlichen Präferenzen zusammenzustellen. Dies erfolgt auf der Grundlage eines eigenen Profils, das Organisationseinheiten (z. B. Lerngruppen) und Themenbereiche registriert. In der folgenden Abbildung wird ein Beispiel für ein Profil („Mein Arbeitsplatz“) dargestellt, das bestimmte Veranstaltungen (z. B. Onlinekurse) umfasst.

Sponsored by Wirtschaftsinformatik and Controlling

Volksbanken Raiffeisenbanken

Mein Arbeitsplatz | Abo | Personliche Daten | Anwenderforum | Logout

Mein Arbeitsplatz: Home

Hallo Conny Controlling! Wir freuen uns, Sie wieder begrüßen zu dürfen.

Alle Funktionen von OpenUSS sind auf der Funktionsleiste auf der linken Seite zu finden. Die Navigationsleiste oben zeigt Ihre momentane Position. Verwenden Sie bitte diese Navigationsleiste, um sich innerhalb von OpenUSS zu bewegen.

Bitte wählen Sie aus den folgenden Veranstaltungen aus. Ebenso können Sie die Institutionen durchsuchen, bei denen Sie sich eingetragen haben.

Abonnierte Veranstaltungen

- [TOP \(Akademie Deutscher Genossenschaften - 2005\)](#)
- [FIT \(Akademie Deutscher Genossenschaften - 2005\)](#)
- [FIT F2 \(Akademie Deutscher Genossenschaften - 2005\)](#)
- [GBF \(Akademie Deutscher Genossenschaften - 2005\)](#)

Ausgewählte Fächer

- [Change Manager ADG \(Akademie Deutscher Genossenschaften\)](#)

Ausgewählte Institutionen

- [Akademie Deutscher Genossenschaften \(1099052588360\)](#)

Suchen

Sie können beispielsweise nach Institution, Institutionsinhaber oder auch nach der Vorlesung bzw. ihrer ID-Nummer suchen.

Suchbegriff:

Einfügen einer Veranstaltungs-Id

Veranstaltungs-Id:

Nachrichten von Institutionen

- Akademie Deutscher Genossenschaften - 09.11.04: Willkommen in ADG! [mehr](#)
- Akademie Deutscher Genossenschaften - 09.11.04: Semesteranfang 2005 [mehr](#)

(c) Copyright OpenUSS 2000 - 2004 devanto@uni-muenster.de
Letzte Änderung: 09.11.04

Abb. 8: Seite „Mein Arbeitsplatz“ mit abonnierten Lehrveranstaltungen

Nach der Auswahl aus dem registrierten Informationsangebot können entsprechende Administrationsfunktionen wahrgenommen werden. Zu diesen Funktionen gehören Kommunikationsfunktionen, wie z. B. der Besuch eines veranstaltungsbezogenen Chat-Bereichs, die aktive oder passive Teilnahme am Diskussionsforum und das Abonnieren eines E-Mail-Verteilers. Darüber hinaus werden nach der Selektion einer Vorlesung automatisch veranstaltungsbezogene Mitteilungen (z. B. Terminänderungen und Themenhinweise) eingeblendet. Auf diese Weise wird für die Lernenden eine homogene Sicht auf sämtliche veranstaltungsrelevanten Informationen erzeugt. Aus der gleichen Sicht kann auch der Zugriff auf die verfügbaren Lernmaterialien der selektierten Veranstaltung vorgenommen werden. Zu jeder vom Dozenten angebotenen Veranstaltung kann der Umfang der Administrationsfunktionen per Customizing zusammengestellt werden. Nicht benötigte Funktionen werden hierdurch deaktiviert.

3.4 Organisation des Entwicklungsprozesses

Bei der Anpassung von OpenUSS für die Domäne *Genossenschaftsbanken* wurde sichergestellt, dass die OpenUSS-Software per Internet distribuiert und kollaborativ weiterentwickelt werden kann. Um eine kollaborative Weiterentwicklung im Rahmen der Open Source-Community zu gestatten, wurden die Entwicklungsplattform SourceForge.net gewählt. Diese Plattform „beherbergt“ über 80.000 Open Source-Projekte und unterstützt über 430.000 registrierte Entwickler. Dazu gehören auch etablierte Open Source-Projekte, wie z. B. KDE, Python und MySQL. Die Auswahl von SourceForge erfolgte vor dem Hintergrund der angebotenen Mehrwertdienste für den Open Source-Entwicklungsprozess. So unterstützt SourceForge die Analyse-, Entwurfs-, Implementierungs- und Testphase sowie die Koordination sämtlicher Teilaktivitäten (Abb. 9). Dabei wird auch ein leistungsfähiges Supportsystem bereitgestellt,

dass die systematische Dokumentation und Verfolgung von Programmfehlern gestattet. Interessierte Entwickler und Anwender können auf diese Weise bereits im Anfangsstadium in den Entwicklungsprozess integriert werden und tragen so zur Qualitätssicherung und Weiterentwicklung im internationalen Kooperationsverbund bei.

Project: Open University Support System: Summary

[Überblick](#) | [Admin](#) | [Homepage](#) | [Tracker](#) | [Bugs](#) | [Support](#) | [Mailinglisten](#) | [Aufgaben](#) | [Dokumentation](#) | [Neuigkeiten](#) | [CVS](#) | [Dateien](#)

This is the first Open Source Project for an administration system aiming to support universities and faculties. Our mission is to establish a standard OpenUSS API (Open University Support System Application Programming Interface).

Java Foundry

- Development Status: 5 - Production/Stable
- Environment: Other Environment
- Intended Audience: Developers, End Users/Desktop
- License: GNU General Public License (GPL)
- Operating System: OS Independent
- Programming Language: Java
- Topic: Education

Project UNIX name: openuss
 Registered: 2000-05-29 01:33
 Activity Percentile (last week): 80.1558%
[View project activity statistics](#)

Entwicklerinformationen

Projektbetreuer:
 dewarito
 awfirbe
 menzel

Entwickler:
 10 [[View Members](#)]

Letzte Dateireleases

Paket	Version	Date	Anmerkungen / Überwachen	Downloaden
openuss-developer-progs	OpenUSS Version 1.2 Final	April 16, 2002	-	Downloaden
openuss-user-progs	OpenUSS Version 1.1	September 12, 2001	-	Downloaden
openuss-developer-docs	OpenUSS Dev 1.0	April 5, 2001	-	Downloaden
openuss-user-docs-manual	OpenUSS Docs Version 1.2	April 25, 2002	-	Downloaden
openuss-user-docs-install	OpenUSS Docs Install	April 26, 2002	-	Downloaden

[\[View ALL Project Files\]](#)

Öffentliche Bereiche
Letzte Neuigkeiten

[Projekthomepage](#)
[OpenUSS 1.2 Final is out!](#)

Abb. 9: OpenUSS-Projektmanagement mit SourceForge

Die Entscheidung zur Einführung und Anwendung von OpenUSS in komplexen Unternehmensorganisationen hängt von einer Vielzahl situativer Einflussfaktoren ab, die mit dem jeweiligen informationstechnologischen Kapazitäts- und Kompetenzprofil der nutzenden Organisationseinheit variieren. Vielfach verfügen Organisationseinheiten nicht über die notwendige Infrastruktur (z. B. ausfallsichere Server) bzw. das notwendige Know-how zur Installation und Wartung. Dies ist insbesondere bei Open Source-Produkten wie OpenUSS der Fall, die die Installation und Konfiguration weiterer Sekundärprodukte (z. B. WWW-Server, Datenbanksoftware) voraussetzen. Diesem Aspekt wurde bei der Entwicklung von OpenUSS Rechnung getragen, indem auch der Betrieb durch einen *Application Service Provider (ASP)* vorgesehen wurde. Auf diese Weise können Rechenzentren eine zentrale OpenUSS-Instanz betreiben, die von mehreren Organisationseinheiten gleichzeitig genutzt wird. Dieser Zentralisierungsansatz trägt dazu bei, den Wartungsaufwand zu senken und verfügbare IT-Infrastrukturen effizient zu nutzen. Diese Effekte werden besonders deutlich, wenn dadurch bestehende heterogene Infrastrukturlandschaften konsolidiert werden können. Dieser Gestaltungsansatz unterstützt auch die Umsetzung von Outsourcingstrategien zur Bereitstellung von E-Learning-Services.

3.5 Wirtschaftlichkeitsbetrachtung

Um die Wirtschaftlichkeit der Einführung und Anwendung einer E-Learning-Plattform wie OpenUSS zu untersuchen, sind die Kosten- und Nutzeneffekte zu quantifizieren. Zwar ist der erzielbare Nutzen nur im jeweiligen Anwendungskontext situativ erfassbar (z. B. mithilfe von Nutzwertanalysen), doch eröffnet die Anwendung quelloffener Software strategische Kostensenkungspotenziale. So impliziert der Begriff Open Source, dass das entsprechende Softwareprodukt ohne Zahlung von Lizenzgebühren über verteilende Institutionen (z. B. Mediatoren) erhältlich ist. Gegenüber kommerzieller Software (i. d. R. Closed Source) weisen Open Source-Produkte einen Kostenvorteil in Bezug auf die zu zahlenden Lizenzgebühren auf, der sich in langfristigen Wirtschaftlichkeitsbetrachtungen positiv niederschlägt. Dabei ist allerdings zu berücksichtigen, dass der Großteil der softwarebezogenen Gesamtkosten (Total Cost of Ownership, TCO) durch den laufenden Betrieb von Softwaresystemen verursacht wird. Infolgedessen sind im Rahmen von Wirtschaftlichkeitsanalysen im Einzelnen die in der folgenden Abbildung gegliederten Kostenarten zu erfassen:

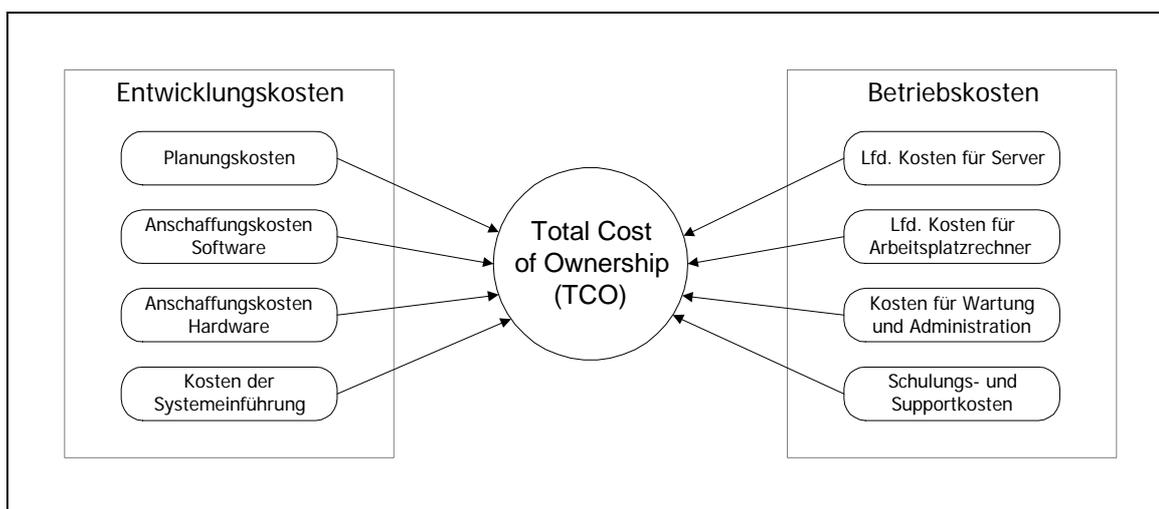


Abb. 10: Gliederung der Entwicklungs- und Betriebskostenarten für E-Learning-Softwareprodukte

In Bezug auf die Entwicklungskosten wirkt sich beim Einsatz von OpenUSS die java-basierte Architektur günstig aus. So ist das System auch auf älteren Hardwarearchitekturen lauffähig und setzt keine proprietären Komplementärprodukte voraus, die zusätzliche Lizenzgebühren verursachen. Da OpenUSS auf (theoretisch) beliebig viele Rechnerknoten verteilt werden kann, können bereits bestehende Rechnersysteme integriert werden, sodass die Skalierungskosten gering gehalten werden. Hieraus resultieren Kostensenkungspotenziale für den Bereich der hardware- und softwarebezogenen Anschaffungskosten. Dabei ist allerdings zu berücksichtigen, dass eine Anpassung der Software, etwa in Bezug auf die Anwendungslogik oder die Benutzeroberfläche, entsprechende Programmierleistungen erfordert, die im Rahmen der Wirtschaftlichkeitsrechnung als Kosten der Systemeinführung zu berücksichtigen sind.

Bezüglich der Betriebskosten ist festzustellen, dass Software-Updates per Internet bezogen und installiert werden können, ohne dass zusätzliche Lizenzgebühren anfallen. Außerdem basiert OpenUSS auf einem Thin Client-Konzept, sodass clientseitig lediglich ein WWW-Browser als Zugangssoftware benötigt wird. Dies macht sich im Systembetrieb vor allem in den Kostenpositionen für Wartung bzw. Administration und den laufenden Kosten für Arbeitsplatzrechner bemerkbar. Darüber hinaus trägt das Open Source-Konzept dazu bei, administrative Kosten des IT-Controllings zu senken. Während beim Einsatz kommerzieller Produkte administrative Tätigkeiten (z. B. Inventarisierung, Abrechnung, technische Vervielfältigungskontrolle) zur Einhaltung der Lizenzbedingungen im Rahmen des Software Asset Managements notwendig sind, fallen diese beim Einsatz von Open Source-Software nicht an.

4 Fazit

Mit der E-Learning-Plattform *OpenUSS für Volksbanken* wird ein systemgestaltender Beitrag zur Unterstützung betrieblicher Personalentwicklungsaktivitäten im Anwendungskontext von Genossenschaftsbanken geliefert. Eine zentrale Eigenschaft dieses Open Source-Softwareprodukts ist seine inhärente Flexibilität. So können durch aufgrund seiner modularen Architektur und der Verfügbarkeit des vollständigen Quellcodes anwendungsfallbezogene Erweiterungen oder Anpassungen vorgenommen werden. Im Rahmen einer ressourcenorientierten Perspektive wird hiermit die informationstechnologische Plattform zur Bereitstellung von E-Learning-Diensten geschaffen. Auf dieser Grundlage können nicht nur innovative Personalentwicklungsstrategien zur Mitarbeiterqualifizierung konzipiert und umgesetzt werden, vielmehr können anhand der Plattform auch die Mitglieder der Genossenschaftsbanken mit E-Learning-Services angesprochen werden. Auf diese Weise können genossenschaftsbezogene Bildungsinhalte effizient disseminiert werden, sodass ein positiver Beitrag zur Mitgliederförderung erzielt wird. Letztlich gehen von dieser Anwendung auch positive Impulse in Bezug auf die Stärkung der genossenschaftlichen Gemeinschaft durch Mitgliederbindung und -gewinnung aus.

Literatur

- Baumgartner, P., Häfele, H., Maier-Häfele, K. (2002), Learning Management Systeme: Ergebnisse einer empirischen Studie – Evaluationsdesign und Auswahlempfehlungen. in: Campus 2002 – Die Virtuelle Hochschule in der Konsolidierungsphase, Hrsg.: G. Bachmann, O. Haefeli, M. Kindt, Münster et al. 2002, S. 287-296.
- Bensberg, F., Dewanto, B. L. (2003), TCO VOFI for eLearning Platforms, in: Poster Abstracts of the 25th International Conference on Information Technology Interfaces (ITI 2003), Hrsg.: SRCE, Zagreb (Kroatien) 2003, S. 9-12.
- Bensberg, F., Dewanto, L. (2001), Wandel im Softwaremarkt durch Open Source, in: Software Development, 1. Jg., 2001, H. 3, S. 51-55.
- Buchholz, H., Leypold, M. E., Schilling, T. (2003), Lehr- und Lernmanagementsysteme im Vergleich – Ein Überblick zur Entscheidungshilfe für das Rechenzentrum der Universität Rostock, Rostock 2003.
- Doberkat, E., Veltmann, C., Engels, G., Hausmann, J. H., Lohmann, M. (2002), Anforderungen an eine eLearning-Plattform – Innovation und Integration, Studie im Auftrag des Ministeriums für Schule, Wissenschaft und Forschung des Landes Nordrhein-Westfalen, Dortmund 2002.
- Eim, A. (2004), Das Drei-Säulen-System der deutschen Kreditwirtschaft unter besonderer Berücksichtigung des Genossenschaftlichen Finanzverbundes, Arbeitspapier Nr. 40 des Instituts für Genossenschaftswesen der Westfälischen Wilhelms-Universität Münster, Münster 2004.
- Friedewald, M., Blind, K., Edler, J. (2002), Die Innovationstätigkeit der deutschen Softwareindustrie, in: Wirtschaftsinformatik, 44. Jg., 2002, H. 2, S. 151-161.
- Grob, H. L., Bensberg, F. (2003), Strategische Potenziale von Open Source-Software für die computergestützte Hochschullehre (cHL), Arbeitsbericht Nr. 24 der Reihe „CAL+CAT“, Hrsg.: H. L. Grob, Münster 2003.
- Grob, H. L., Bensberg, F., Dewanto, B. L. (2004), Developing, Deploying, Using and Evaluating an Open Source Learning Management System, in: Proceedings of the 26th Int. Conf. Information Technology Interfaces ITI 2004, Zagreb (Croatia) 2004, S. 387-393.
- Grob, H. L., Bensberg, F., Dewanto, B. L., Düppe, I. (2004), Controlling von Learning Management-Systemen – ein kennzahlenorientierter Ansatz, in: Campus 2004 (Tagungsband), Kommen die digitalen Medien an den Hochschulen in die Jahre?, Hrsg.: D. Carstensen, B. Barrios, Band 29 der Gesellschaft für Medien in der Wissenschaft (GMW), Münster 2004, S. 46-56.

- Grob, H. L., Bensberg, F., Dewanto, L. (2001), Das cHL-Administrationssystem OpenUSS, in: cHL: computergestützte Hochschullehre – Dokumentation zum cHL-Tag 2000, Hrsg.: H. L. Grob, Münster et al. 2001, S. 31-40.
- Grob, H. L., Bensberg, F., Kaderali, F. (2004 a), Controlling Open Source Intermediaries – a Web Log Mining Approach, in: Proceedings of the 26th Int. Conf. Information Technology Interfaces ITI 2004, Zagreb (Croatia) 2004, S. 233-242.
- Grob, H. L., Bensberg, F., Kaderali, F. (2004 b) A Behavioural Approach for the Coordination of Open Source Software Intermediaries, in: INFORMATIK 2004 – Informatik verbindet, Band 2, Beiträge der 34. Jahrestagung der Gesellschaft für Informatik e.V. (GI), Ulm, 20.-24. September 2004, Hrsg.: P. Dadam, M. Reichert, Bonn 2004, S. 504-508.
- Grob, H. L., Lahme, N. (2004), Total Cost of Ownership-Analyse mit VOFI, Arbeitsbericht Nr. 20 der Reihe „Computergestütztes Controlling“, Hrsg.: H. L. Grob, Münster 2004.
- Grob, H. L., Reepmeyer, J.-A., Bensberg, F. (2004), Einführung in die Wirtschaftsinformatik, 5., vollst. überarb. u. erw. Aufl., München 2004.
- Hettrich, A., Koroleva, N. (2003), Learning Management Systeme (LMS) und Learning Content Management Systeme (LCS) – Fokus Deutscher Markt, Stuttgart 2003.
- IEEE (2001), P1484.1/D9, Draft Standard for Learning Technology — Learning Technology Systems Architecture (LTSA), New York 2001.
- Koordinierungs- und Beratungsstelle der Bundesregierung für Informationstechnik in der Bundesverwaltung im Bundesministerium des Inneren (KBSt) (Hrsg., 2001), WiBe 21 – Empfehlung zur Durchführung von Wirtschaftlichkeitsbetrachtungen in der Bundesverwaltung, insbesondere beim Einsatz von IT (Schriftenreihe, Band 52). Berlin 2001.
- Köster, H. (2001), CampusSource – Eine Initiative für die Hochschulinfrastrukturen in: cHL: computergestützte Hochschullehre – Dokumentation zum cHL-Tag 2000, Hrsg.: H. L. Grob, Münster et al. 2001, S. 19-30.
- Lindemann, J. (2004), Das Kreditgeschäft der Genossenschaftsbanken im Wandel – Strategische Herausforderungen und operative Handlungserfordernisse einer risikodifferenzierten Bepreisung im Kreditgeschäft, in: ADG Schriften, Bd. 1, Montabaur 2004.
- Ljungberg, J. (2000), Open Source Movements as a Model for Organising, in: European Journal of Information Systems, 9. Jg., 2000, H. 4, S. 208-216.

- Lutterbeck, B., Gehring, R., Horns, A. (2000), Sicherheit in der Informationstechnologie und Patentschutz für Software-Produkte – ein Widerspruch?, Berlin 2000.
- Müller-Lietzkow, J. (2004), Free Resources and the Resource-Based View, in: INFORMATIK 2004 – Informatik verbindet, Band 1, Beiträge der 34. Jahrestagung der Gesellschaft für Informatik e.V. (GI), Ulm, 20.-24. September 2004, Hrsg.: P. Dadam, M. Reichert, Bonn 2004, S. 493-498.
- Naumann, S., Schmitz, A. (2004), Evaluation zu Einsatz, Entwicklung und Bewertung von Informationstechnik in selbstorganisierten Gemeinschaften und Gruppen, Arbeitsbericht Nr. 1 des Instituts für Softwaresysteme in Wirtschaft, Umwelt und Verwaltung (ISS), Birkenfeld 2004.
- O'Reilly, T. (1999), Lessons from Open-Source Software Development, in: Communications of the ACM, 42. Jg., 1999, H. 4, S. 33-37.
- Pearson, H. (2000), Open Source Licenses, in: Computer Law & Security Report, 16 (3), S. 151-156.
- Rahmen-Zurek, K. (2000), Internet-Ökonomie und Genossenschaftswesen – eine organisationstheoretische Betrachtung, in: Internet – Chancen für Genossenschaften, Hrsg.: T. Theurl, Münster 2000, S. 5-44.
- Rost, J. (1997), Wiederverwendbare Software, in: Wirtschaftsinformatik, 39. Jg., 1997, H. 4, S. 357-365.
- Sandl, U. (2001), Open Source-Software: Politische, ökonomische und rechtliche Aspekte, in: Computer & Recht, 17. Jg., 2001, H. 5, S. 346-351.