

Advanced Topics of Information Science

4. Open Source – Integration im Unternehmen

Prof. Dr. habil. Wolfgang Semar (wolfgang.semar@htwchur.ch)

Hochschule für Technik und Wirtschaft Chur
Schweizerisches Institut für Informationswissenschaft



[Creative Commons Attribution-NonCommercial-ShareAlike 2.5 Switzerland License](#)

Inhalt

- Stärken und Schwächen von OSS (SWOT-Analyse)
- OSS im Rahmen der Softwareauswahl
- Faktoren für den Erwerb und Entwicklung von OSS
- OSS-Portale
- OS-Wettbewerber
- OSS-Entscheidung: Kosten, Nutzen und strategische Aspekte
- Umsetzung: von der Entscheidung zur Migration

Stärken und Schwächen von OSS

- **SWOT-Analyse**
 - Strengths (Stärke)
 - Weaknesses (Schwäche)
 - Opportunities (Potenziale)
 - Threads (Bedrohungen)

Stärken und Schwächen von OSS

- **SWOT-Analyse**
- **Strengths (Stärke)**
 - Offenheit des Quellcodes
 - Lerneffekt
 - Sicherheit
- **Weaknesses (Schwäche)**
 - OSS ist meist nicht 100% passend
 - Anpassungskosten
 - Supportvertrag
 - Qualität der Software?

Stärken und Schwächen von OSS

- Opportunities (Potenziale)
 - Kosten
 - Kopien sind kostenlos
 - Microsoft hat Kostenvorteil bis 50 Mitarbeiter
 - Ab 250 Mitarbeiter ist Windows 3mal so teuer wie Linux
 - [linux_vs_windows_pricing_comparison.pdf](#)
 - Herstellerunabhängigkeit
 - Eigene Softwareentwicklungsmöglichkeit und somit eigener Innovationszyklus
 - PR-Gründe
 - Clientel
 - Standards setzten sich schneller durch (Java von Sun)

Stärken und Schwächen von OSS

- Threads (Bedrohungen)
 - Mangel an OSS-Kompetenz im Unternehmen
 - Forking (http://www.dwheeler.com/oss_fs_why.html#forking)
 - OSS-Programmierer sind „frei“ im Geiste und in der „Zeitplanung“
 - Starre Unternehmensstrukturen sind meist kontraproduktiv
 - Unsichere Rechtslage
 - Copyright
 - Patente
 - Gewährleistung
 - Haftung
 - Urheberschaft

OSS im Rahmen der Softwareauswahl

- Wann soll OSS bei der Auswahl von Software in Erwägung gezogen werden?
 -

OSS im Rahmen der Softwareauswahl

- Wann soll OSS bei der Auswahl von Software in Erwägung gezogen werden?
 - Bei der Frage ob Zukauf oder Eigenentwicklung
 - Kosten, Aufwand, Nutzen
 - Bei strategischen Wettbewerbsvorteilen
 - Kein strategischer Wettbewerbsvorteil

OSS im Rahmen der Softwareauswahl

- Bei **strategischen Wettbewerbsvorteilen** (gegenüber der Konkurrenz) durch Eigenentwicklungen ist zu berücksichtigen, dass OSS
 - **nur bedingt einsatzfähig** (lizenzabhängig!) ist
 - aber auch als „zusätzliches Geschäftsmodell“ denkbar ist!
- Externer Bezug individuell erstellter Software möglicherweise auch problematisch
 - Geheimhaltung gegenüber Konkurrenten
 - zudem treten Transaktionskosten auf für Koordination, Überwachung und Vertragsgestaltung.

OSS im Rahmen der Softwareauswahl

- Ist **kein „strategischer Vorteil“** vorhanden:
 - Stehen als Entscheidungskriterien nur die „Kosten“ (Transaktions-, Erstellungs-, Anpassungs-) an.
 - Veröffentlichung der angepassten OSS führt zur Verbesserung der Software
 - Weniger Abhängigkeit von einem „Hersteller“
 - Einsehbarkeit in den Quellcode erhöht die Sicherheit
 - Bei OSS fallen keinerlei Kosten für Lizenzierung oder sonstige Transaktionskosten an.
 - Nachteile von OSS?

Erwerb und Entwicklung von OSS

- Erwerb
 - Die „reine“ Nutzung stellt absolut kein Problem dar, solange man die Software nicht verändern möchte.
 - Kostenvorteil gegenüber kommerzieller Software
 - Problem der Softwarepflege
 - Haftungsprobleme
 - Eigene Testdurchläufe sind notwendig
- Entwicklung
 - Kenntnisse über die Lizenzen sind notwendig
 - Kombinierbarkeit
 - Veröffentlichungspflicht

Open Source Portale

- Für die OSS-Entwicklung sind **Portale** sehr hilfreich, da die zahlreichen Programmierer weltweit verteilt sind.
 - **Bazaar vs. Cathedrale**
 - The Cathedral and the Bazaar: Musings On Linux And Open Source By An Accidental Revolutionary von **Eric S. Raymond** von O'Reilly Media
 - <http://web.archive.org/web/20060501180017/http://www.linux-magazin.de/Artikel/ausgabe/1997/08/Basar/basar.html>
 - **Bazaar**
 - Code ist immer offen einsehbar
 - Keine Hierarchie bei den Programmierern
 - Viele Schultern tragen das Projekt

Open Source Portale

- Für die OSS-Entwicklung sind **Portale** sehr hilfreich, da die zahlreichen Programmierer weltweit verteilt sind.
 - Portale haben folgende **Vorteile**:
 - Zentrale Ablage für den Code und Dokumente
 - Einheitliche Mechanismen für das „Konflikt-Management“
 - Infrastruktur für die Kommunikation der Entwickler
 - Verwaltung von Fehlermeldungen
 - Bereitstellung von Tools (Entwicklung, Mailinglisten, Foren)
 - Entwickler, Anwender, Distributoren und andere Beteiligte der Wertschöpfungskette treffen sich zentral

Open Source Portale

- **BerliOS** (wird letztendlich von der BRD gefördert)
 - <http://www.berlios.de/>
 - OSS-Plattform für:
 - Information
 - Entwicklung; „**Developer**“-Portal
 - Präsentation
 - Kostenlose Teilnahme

Open Source Portale

- **SourceForge** (kommerziell)
 - <http://www.sourceforge.net>
 - Plattform für Entwickler (und Anwender)
 - Wird von Open Source Development Network (OSDN) betrieben (Tochter von VA Software)
 - **Collaborative Development System (CDS)** bietet spezielle Dienste an wie z.B.
 - Compiler Farm mit Rechnern unterschiedlicher Betriebssysteme
 - MySQL-Datenbank für Testzwecke

Open Source Portale

- **OW2 Forge** (ehemals ObjectWeb Forge, ObjectWeb and Orientware)
 - <http://forge.ow2.org/>
 - Plattform für Entwickler von „Middleware“ (und Anwender)
 - Wird von Debian unterstützt
- **Savannah** (Free Software Foundation)
 - <http://savannah.gnu.org/>
 - Für GNU-SW
 - Basiert auf der gleichen, aber „freien“ SW wie SourceForge (was sonst!)
 - <http://savannah.nongnu.org/>

OSS-Entscheidung

- Faktoren, die die Entscheidung (pro/contra OSS) beeinflussen anhand der **TOC-Methode (Total Cost of Ownership)**:
 - Monetäre Faktoren
 - Direkte Kosten
 - Indirekte Kosten
 - Nicht monetäre Faktoren
 - Qualitätsaspekte
 - Strategische Aspekte

OSS-Entscheidung

- Direkte Kosten
 - Lizenzkosten
 - Hardwarekosten
- Indirekte Kosten
 - Anpassbarkeit der Software
 - Administration
 - Know-How
 - Support und Updates
 - Integration und Migration

OSS-Entscheidung

- Qualitätsaspekte
 - Stabilität
 - Sicherheit
 - Nutzerfreundlichkeit
 - Skalierbarkeit und Performanz
- Strategische Aspekte
 - Unabhängigkeit
 - Wettbewerb bei Hard- und Software
 - Verfügbarkeit von Anwendungen

OSS-Entscheidung - Direkte Kosten

- Lizenzkosten
 - Bei sinkenden Hardwarekosten fallen die Lizenzkosten immer mehr ins Gewicht
 - Folgende Fragen muss man sich stellen:
 - Sind die Lizenzkosten von OSS tatsächlich und dauerhaft deutlich niedriger als die von proprietärer Software?
 - In welchen Bereichen des Softwareeinsatzes können Lizenzkostenvorteil so bedeutend sein, dass sie eventuelle Nachteile bei anderen Faktoren aufwiegen?
 - Braucht man einen Support und/oder Wartung?

OSS-Entscheidung - Direkte Kosten

- Lizenzkosten
 - Support/Wartung gewünscht?
 - Regelmässige Updates
 - Beseitigung sicherheitskritischer Fehler
 - Kommerzielle Lizenz erwünscht?
 - Software wird oft unter einer dualen Lizenz veröffentlicht
 - Bei der GPL müssen Änderungen wieder unter GPL gestellt werden
 - Wer das nicht haben möchte muss eine kommerzielle Lizenz erwerben
- Embedded OSS
 - Kombination mit proprietärer und spezieller SW

OSS-Entscheidung - Direkte Kosten

- Lizenzkosten
 - Konkurrenz durch OSS drückt ebenfalls den Preis von proprietärer SW
 - Überlegungen zum Wechsel können auch als Verhandlungshebel genutzt werden.
 - Lizenzkosten proprietärer SW hängt oft auch von der Anzahl der Clients ab

OSS-Entscheidung - Direkte Kosten

■ FAZIT

- Kosteneinsparungen durch niedrige Lizenzkosten von OSS sind umso grösser, je:
 - geringer der Bedarf an Zusatzpaketen oder -dienstleistungen wie Support, Updates oder spezifischen Nutzungsrechten ist,
 - teurer alternative proprietäre Lösungen sind (z.B. bei Serverlizenzen),
 - weniger stark Gebühren für OSS mit der Zahl der eingesetzten Kopien steigen,
 - mehr Kopien der OSS eingesetzt werden sollen.

OSS-Entscheidung - Direkte Kosten

■ Hardwarekosten

- Ablösung von teuren Hochleistungsservern (RISC-Server und proprietäres Unix) durch Standardhardware und Linux
 - Anschaffungskosten und Supportkosten
- Ablösung durch einzelne Rechner oder Cluster?
- Laufende Wartungs- und Pflegekosten der Hardware?
- Längere Nutzung bestehender Hardware durch Einsatz von (effizienteren) OSS-Betriebssystemen
- Bessere Ausnutzung bestehender HW durch OSS (Zusammenführung von Diensten)

OSS-Entscheidung - Direkte Kosten

- FAZIT

- Hardwarekosten lassen sich potenziell einsparen durch:
 - den Wechsel von proprietären Hochleistungsservern auf preiswerte Standardhardware
 - längere Nutzung von Systemen aufgrund geringerer Ressourcenanforderungen von OSS
 - Konsolidierung von Systemen auf der Basis von OSS

OSS-Entscheidung - Indirekte Kosten

- Anpassbarkeit der Software
 - Der offene Sourcecode erlaubt eine unmittelbare Veränderung der Software selbst und ihrer Fähigkeiten
 - Der (meist) modulare Aufbau von OSS und die Orientierung an offenen Standards erlauben die optimale Zusammenstellung der für eine Aufgabe benötigten Komponenten (ohne Veränderung des Codes)
 - Code der im Bazaar erstellt wird muss modularer sein!
 - Dadurch leichtere Anpassbarkeit
 - Interoperabilität durch Standards

OSS-Entscheidung - Indirekte Kosten

■ FAZIT

- Die Möglichkeit zur Veränderung des Source Codes bei OSS ist besonders bedeutend, wenn:
 - eigene SW interoperabel mit OSS sein soll und durch eigenes Mitwirken am Entwicklungsprozess die SW-Qualität verbessert werden kann.
 - ein Unternehmen im Fall der Einstellung der Weiterentwicklung von SW die Möglichkeit haben möchte, die SW selbst weiter zu entwickeln.

OSS-Entscheidung - Indirekte Kosten

■ FAZIT

- Die Modularität von OSS ist besonders vorteilhaft, wenn:
 - aufgrund von Hardwarebeschränkungen schlanke SW gefragt ist,
 - die SW exakt an die Anforderungen bestimmter Nutzungen angepasst werden soll, um Sicherheitsprobleme oder erwünschte Nutzungen auszuschliessen.

OSS-Entscheidung - Indirekte Kosten

- Administrationskosten
 - Geringerer oder höherer Administrationsaufwand?
 - ▣ Muss individuell geprüft werden.
- ▣ Know-How-Kosten
 - ▣ Intern verfügbares oder leicht zugängliches Know-How
 - ▣ Nutzung
 - ▣ Administration
 - ▣ Schulung
 - ▣ Personal auf dem Arbeitsmarkt?

OSS-Entscheidung - Indirekte Kosten

- Support- und Update-Kosten
 - Kosten bei der Unterstützung in Problemsituationen
 - Gewährleistung
 - Support
 - Update
 - Wartung
 - Telefonsupport
 - Internet-Foren
 - Zertifizierungen

OSS-Entscheidung - Indirekte Kosten

- Integrations- und Migrationskosten
 - Umstellung in Teilbereiche und schrittweise Migration oder „All in One“-Lösung?
 - Nutzung alter Programm-Teile?
 - Migration der Nutzer!

OSS-Entscheidung - Qualitätsaspekte

- Stabilität
 - Ausfallkosten, Ausfallwahrscheinlichkeit
 - Die Stabilität hängt vom schwächsten Glied ab!
- Sicherheit
 - Virenattacken
 - Programmcode ist offen
 - Qualität des Codes?
 - Sicherheitslücken werden von der Community erkannt

OSS-Entscheidung - Qualitätsaspekte

- Nutzerfreundlichkeit
 - Wird zunehmend besser
- Skalierbarkeit und Performanz
 - Ist eine Frage des eigenen Bedürfnisses
 - OSS kann durchaus performanter sein

OSS-Entscheidung – Strategische Aspekte

- Unabhängigkeit und stärkerer Wettbewerb für Hardware und Software
 - Unabhängigkeit gegenüber den Softwareanbietern
 - Unabhängigkeit gegenüber Hardware da oft plattformübergreifende Lösungen
 - Offene Standards und standardisierte Schnittstellen
- Verfügbarkeit der Anwendungen
 - Linux wird in der Zwischenzeit von vielen (proprietären) Business-Anwendungen unterstützt.

Umsetzung - Von der Entscheidung zur Migration

- Zum grössten Teil ist die Einführung von OSS ein IT-Projekt wie jedes andere auch.
- Die Einführung von OSS ist meist eine Migration (Ablösung vorhandener Systeme)
- Auf Endnutzerbedürfnisse ausreichend Rücksicht nehmen
 - Nicht mal „schnell installieren, es kostet ja nichts“!
 - Endnutzer in die Planung mit Hilfe von „Business Cases“ einbeziehen

Umsetzung - Von der Entscheidung zur Migration

- Die Vorbereitung und Durchführung besteht aus 2 Teilen:
 1. Adoptionsentscheidung (Entscheidung über den Einsatz von OSS)
 2. Die eigentliche Migration

Umsetzung - Von der Entscheidung zur Migration

1. Adoptionsentscheidung

- Analyse der Ist-Situation,
- Identifikation potenzieller Einsatzbereiche,
- Entwicklung des Business Case
- Weiteres Vorgehen

Pilotprojekt(e)

Auswertung der Pilotprojekte,
Entscheidung über Migration

- Kenntnisse lizenzrechtlicher Aspekte sind notwendig
- Auswahl eines sinnvollen Pilotprojekts (Umfang, Komplexität, Zielkongruenz, aufgeschlossene Pilotnutzer, Laufzeit,..)
- Scheitern des Pilots muss genau untersucht werden
- Festlegen der Eckdaten für eine Migration (Kosten,..)

Umsetzung - Von der Entscheidung zur Migration

2. Migration

Feinplanung
der Migration

Entwicklung und
Beschaffung v.
Komponenten

Roll-Out

Betrieb

- Meilensteine und Abschlusskriterien festlegen
- Training nicht vergessen
- Parallelbetrieb von Alt und Neu
- Fortlaufende Kontrolle

Vielen Dank für Ihre Aufmerksamkeit.



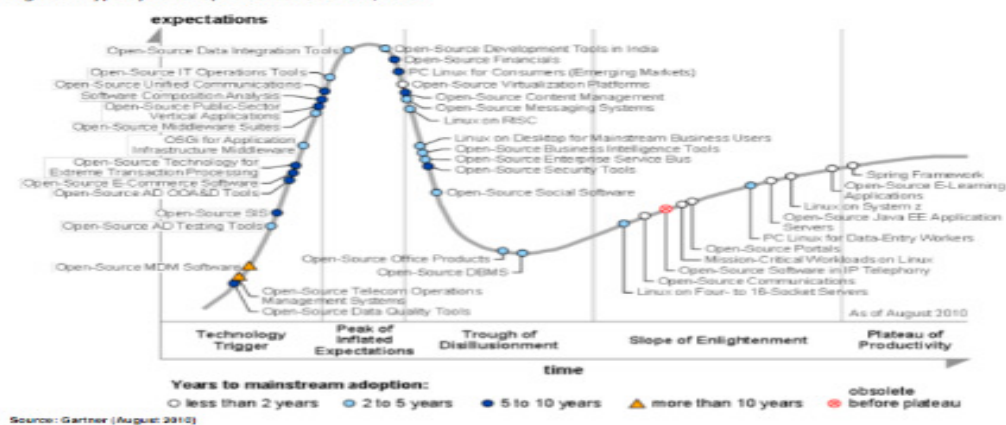
Prof. Dr. Wolfgang Semar (wolfgang.semar@htwchur.ch)

Hochschule für Technik und Wirtschaft Chur
Schweizerisches Institut für Informationswissenschaft

www.informationswissenschaft.ch

Hype Cycle for Open-Source Software, 2007" report,

Figure 1. Hype Cycle for Open-Source Software, 2010



Publication Date: 10 August 2010/ID Number: G01203556
© 2010 Gartner, Inc. and/or its Affiliates. All Rights Reserved.

Page 6 of 65
Gartner

<http://www.xwiki.com/xwiki/bin/view/Blog/XWiki+Hype+Cycle+Open+Source+2010>

http://www.gartner.com/it/content/1395400/1395423/august_4_whats_hot_hype_2010_ifenn.pdf

Fazit

Open-Source-Software ist per se nicht
schlecht!

Man muss sie nur zulassen, dann
entwickeln sich auch tragfähige
Geschäftsmodelle.