

## Fallstudie: Wikis und Weblogs bei Sun Microsystems

*Von Wolf Hilzensauer, Sandra Schaffert*

Die Anwendungen von Social Software im betrieblichen Umfeld sind vielfältig: Sie reichen vom individuellen Ideenmanagement, der Koordination und Dokumentation gemeinsamer Entwicklungsarbeiten (Projektmanagement) bis zum Customer-Relationship-Management [vgl. John et al. 2005]. Dieser Beitrag stellt ein Good-Practice-Beispiel dar, wie Social Software in der Firma Sun Microsystems prozess- und projektübergreifend erfolgreich eingesetzt wird und die Firmenphilosophie gleichermaßen trägt wie bestimmt.

Die folgenden Ausführungen beruhen auf den Selbstdarstellungen von Sun Microsystems in unterschiedlichen Berichten auf der Homepage (Stand Oktober 2007), Interviews mit Josef Holy (Java-NetBeans Zentrum in Prag) sowie ergänzenden Recherchen. Insbesondere bei Josef Holy bedanken wir uns für seine Ausführungen, die uns einen Einblick in seine tägliche Arbeit gewährt haben.

Salzburg Research ist die Forschungsgesellschaft des Landes Salzburg und betreibt Forschung und Entwicklung im Bereich der Informations- und Kommunikationstechnologien (IKT). Innerhalb der Salzburg Research beschäftigt sich das Anwendungsfeld „Education and Media“ mit webbasierten Bildungsinnovationen, selbstorganisiertem, technologieunterstütztem Lernen sowie der Entwicklung von innovativen didaktischen Methoden für den Einsatz von Informations- und Kommunikationstechnologien im Bildungskontext. Die Einführung semantischer Technologien und Social Software stellt für uns einen wesentlichen Beitrag für das Lernen im Unternehmen dar. In diesem Artikel verbinden wir das Interesse an der Darstellung des Einsatzes von Social Software in Unternehmen mit dem Umstand, in der Firma Sun Microsystems einen Kooperations- und Projektpartner gefunden zu haben, mit dem wir aktiv an Fortentwicklungen der Wiki-Technologien arbeiten.

### 1 Das Unternehmen: Sun Microsystems

Das Credo von Sun Microsystems ist es, eine technologische Welt ohne technische Hürden zu erschaffen. Schon seit 1982 (also noch vor der Entwicklung des World Wide Webs) verfolgt Sun Microsystems die Vision, eine für alle zugängliche digitale Welt zu schaffen. Zu diesem Zweck entwickelt Sun Computer, Netzwerk- und Speicherkomponenten sowie Programmierumgebungen, die es ermöglichen, unterschiedliche Systemframeworks miteinander zu kombinieren. Zu den bekanntesten Systemen gehören die Java-Technology-Plattform, Solaris Betriebssysteme, StorageTek Speicherkomponenten und UltraSPARC Prozessoren.

Neben anderen Standorten unterhält Sun Microsystems in Prag/Tschechien das Zentrum für die Entwicklung der Java NetBeans IDE. Dieses Entwicklerzentrum beschäftigt derzeit (September 2007) ca. 150 Mitarbeiter, die hauptsächlich an der Weiterentwicklung der Entwicklungsumgebung für Java arbeiten. NetBeans wurde für die Programmiersprache Java entwi-

ckelt, es werden aber auch Sprachen wie C, C++, Ruby und Fortran unterstützt<sup>1</sup>. Die einzelnen Bausteine der NetBeans werden in unterschiedlichen Abteilungen entwickelt, die zu einem Großteil unabhängig von einander agieren. Java NetBeans wird als Open-Source-Produkt entwickelt und kann von den Entwicklerseiten im Internet jederzeit kostenfrei heruntergeladen werden. Alle Mitarbeiter bedienen sich unterschiedlicher Werkzeuge und Orte für die Dokumentation der laufenden Entwicklungen. Hier stand uns Josef Holy als Experte und Ansprechpartner zur Verfügung.

Das gesamte Unternehmen beschäftigt weltweit ca. 35.000 Mitarbeiter und ist in über 100 Ländern der Welt vertreten. Die Zentrale ist im Silicon Valley (Santa Clara/Kalifornien, USA) beheimatet.

Es gibt sicherlich viele Begriffe, mit der sich die Unternehmensphilosophie von Sun Microsystems beschreiben lässt. Auf einem abstrakten Niveau erscheint uns „Offenheit“ als ein wesentliches Kennzeichen des unternehmerischen Handelns: Sun Microsystems unterscheidet sich wesentlich von herkömmlichen Unternehmen, die ein Produkt im Stillen entwickeln, „fertig“ auf den Markt bringen, durch Patente schützen und durch den Vertrieb des Produktes ihr Einkommen erwirtschaften. In einer Studie von Berlecon werden diese Unternehmen als „Enterprise 1.0“ bezeichnet und ihnen zudem folgende Merkmale zugeschrieben [Berlecon 2007]: Hierarchische Strukturen, festgelegte Kommunikation, statisches Wissen, isolierte Wissensinseln, Wissensüberlauf, zentralisierte Information, monolithische Softwareprodukte sowie vierteljährliche Releases (gemeint: neue Versionen der Software). Demgegenüber zeichnet sich ein „Enterprise 2.0“ durch folgende Eigenschaften aus: Projektarbeit, Mitarbeiterbeteiligung, organische Wissensentwicklung, personalisiertes Wissen, offenes und vernetztes Wissen, Innovation durch Kombination, Service-orientierte Architektur, und die andauernde „Betaversion“ („*the perpetual beta*“) [vgl. Berlecon 2007]. Wie sich in den folgenden Abschnitten zeigen wird, ist Sun dieser „Enterprise 2.0“ Gruppe zugehörig. Die aktive Förderung von Open Source sowie die Community-Einbindung sind zwei Aspekte, die wir zunächst vorstellen, bevor wir uns einer konkreten Toolnutzung widmen.

## 2 Die Ausgangssituation: Open Source als Unternehmensphilosophie

Die ersten Entwicklungen von Sun Microsystems wurden bereits 1984 unter einer Open-Source-Lizenz veröffentlicht<sup>2</sup>. Die Entscheidung, das Network File System (und in der Folge die meisten weiteren Produkte) mit offenen, frei verfügbaren und nutzbaren Quellcodes freizugeben, bezeichnet Sun im Rückblick als wesentlichen Beitrag für die erfolgreiche weltweite Entwicklung und den großen Umfang, den die unterschiedlichen Code-Bibliotheken bisher angenommen haben.

Open-Source-Software zu entwickeln bedeutet auch, von herkömmlichen Produktions- und Vermarktungsprozessen abzuweichen. Der Unternehmenserfolg wird nicht durch die direkte Vermarktung von Produkten erreicht, sondern durch das Bereitstellen von zusätzlichen Services wie Beratungsdienstleistungen oder Hardwarekomponenten. Die Veröffentlichung

---

<sup>1</sup> [de.wikipedia.org/w/index.php?title=NetBeans&oldid=42774635](http://de.wikipedia.org/w/index.php?title=NetBeans&oldid=42774635)

<sup>2</sup> [de.wikipedia.org/w/index.php?title=Sun\\_Microsystems&oldid=42431005](http://de.wikipedia.org/w/index.php?title=Sun_Microsystems&oldid=42431005)

von Firmeneigentum und -wissen im Entwicklungsstadium bedeutet darüber hinaus, auf allen Ebenen innerhalb einer Firma eine große Risikobereitschaft einzugehen. Erfolg oder Misserfolg liegen nicht unbedingt unter der Kontrolle oder dem Einfluss von einzelnen Abteilungen oder Arbeitsgruppen. Es gilt, die Entwicklergemeinschaft am Leben zu erhalten, sie mit Informationen und Lösungsstrategien zu versorgen und sie kontinuierlich über die aktuellen internen Entwicklungen zu informieren.

Der Einsatz von Open-Source-Software unterstützt zunehmend auch herkömmliche Bereiche des Softwareengineerings. Es zeigt sich, dass die Übertragung der Open-Source-Idee (wie z.B. die intensive Nutzung kollaborativer Werkzeuge) auf die verschiedenen Entwicklungsprozesse proprietärer Softwareprojekte positiven Einfluss hat [vgl. Koppány und Lehnert 2005].

Sun reagiert nicht nur auf die Folgen der Open-Source-Entwicklung, sondern versteht die Open-Source-Idee und das aktive Community-Management als Teil der Unternehmensstrategie. Darunter wird der Aufbau, die Pflege und die Nutzung von Kunden- bzw. Entwicklergemeinschaften verstanden. Diese Gemeinschaften (auch außerhalb des Unternehmens) interessieren sich für ein gemeinsames Thema (beispielsweise eine bestimmte Software) und verfolgen die Entwicklung mehr oder weniger aktiv. In die Aufrechterhaltung dieser Prozesse wird bei Sun verhältnismäßig viel Energie gesteckt, weil aus der Firmengeschichte bekannt ist, dass eine bestmögliche Unterstützung der Community den größten Mehrwert für das Unternehmen und die Erreichung der strategischen Ziele darstellt.

Die Einbindung der Community erfolgt schon in der Planungsphase, Prototypen werden schon in frühen Projektphasen an die Endbenutzer, also die Lead-User und Early-Adopters, ausgehändigt. Die Entwicklungs-Communities sind von einem schnelllebigen und enormen Innovationspotenzial geprägt. Viele Ideen werden als Ad-Hoc-Entwicklungen in die laufenden Entwicklungen eingebaut und maßgeblich von der Basis („bottom up“), also von den Programmierern, getrieben.

Die Firmenkultur wurde durch den Einsatz von Open-Source-Entwicklung geprägt. Dies geschah im Hinblick auf die Bereitschaft, Informationen und Wissen zur Verfügung zu stellen. Die Open-Source-Philosophie wird bei Sun daher nicht nur als Prinzip der Software-Entwicklung eingesetzt, sondern wird gleichsam als Unternehmensphilosophie verstanden.

### **3 Das Projekt: Wikis für Wissensmanagement in der Softwareentwicklung**

Social Software spielt eine große Rolle bei Sun Microsystems. Schon früh wurde das Potenzial dieser Art von Software erkannt und für interne und externe Wissensmanagementprozesse eingesetzt. Im Folgenden möchten wir das am Beispiel des Wissensmanagements in der Softwareentwicklung zeigen.

#### **3.1 Das Projektziel: Wikis als Werkzeuge für Wissensmanagement**

Durch die vergleichsweise lange Tradition der freien Veröffentlichung von Programmcodes hat Sun bereits im Jahr 1994 die Notwendigkeit des Community-Managements erkannt. Nicht nur interne, sondern auch externe Programmierer arbeiten an der Weiterentwicklung

der verschiedenen Software und es bedarf daher einer leicht zugänglichen Kommunikationsschnittstelle für all diese Programmierer. Seit dieser Zeit setzt die Firma auf unterstützende Software Tools, die die Unternehmensphilosophie nicht nur unterstützen, sondern auch widerspiegeln. Einer der ganz großen Entwicklungen dabei war beispielsweise die Einführung einer zentralen Website für alle Mitarbeiter-Weblogs im Jahr 2003<sup>3</sup>.

Auch (oder vor allem) Wikis werden seit langer Zeit in der Softwareentwicklung eingesetzt und in den verschiedenen Abteilungen auch entsprechend zur Verfügung gestellt. Seit Mai 2007 ist (zusätzlich zu den vielen internen Wikisystemen) ein zentrales Wiki-System im Einsatz<sup>4</sup>, welches als „Enterprise-Wiki“ unter dem Namen „Confluence“ der Firma Atlassian<sup>5</sup> bekannt ist. Dieses Wiki-System schafft die Möglichkeit, dass jeder Mitarbeiter der Firma Sun einen persönlichen Arbeitsbereich (Space) anlegen und sowohl mit Inhalten, als auch mit individuellen Rechten versehen kann.

Für die bessere Durchsuchbarkeit der Wiki-Spaces werden Klassifizierungsmerkmale, so genannte Labels vergeben, die (ähnlich wie Tags bei Weblogs) durchsuchbar sind. Dabei bieten zusätzliche Funktionen, wie etwa „most popular labels“ oder „recent labels“, eine zusätzliche Möglichkeit, sich in den Wiki-Spaces zurecht zu finden.

Es wird deutlich, dass die Verwendung der Wikis bei Sun nicht nur einen Mehrwert für die interne Dokumentation der Softwareentwicklung darstellt. Die Einbindung externer Mitarbeiter bietet die Möglichkeit, über ein zentrales System alle Beteiligten mit gleichen oder ähnlichen Interessen mit Informationen zu versorgen.

### 3.2 Die Durchführung: Wiki als Werkzeug zur Dokumentation von Wissen

Schon seit langem verwenden (vor allem technikaffine) Mitarbeiter bei Sun diverse kollaborative und webbasierte Systeme zur Dokumentation von Softwareentwicklungsprozessen. Die unter dem Schlagwort Social Software bekannten Werkzeuge erfreuen sich auch hier größter Beliebtheit, vor allem, da das Unternehmen die Mitarbeiter zur Verwendung ermutigt. Für die Dokumentation des Softwareentwicklungsprozesses sind bewährte Systeme wie (z.T. öffentliche) Diskussionsforen und Mailinglisten zwar nach wie vor im Einsatz, die Verwendung von Wiki-Systemen setzt sich aber mehr und mehr durch.

Unser Ansprechpartner bei Sun, Josef Holy, konnte leider nicht eruieren, seit wann und in welchem Ausmaß Wikis bei Sun derzeit eingesetzt werden. Er erklärt dies folgendermaßen: Bei Sun gibt es keine zentrale Einheit oder Abteilung, die sich um Wikis oder deren Einsatz kümmert. Auch ist diese gar nicht notwendig, da die Firma voller „Technikexperten“ ist, die keine Schwierigkeiten haben ein Wiki aufzusetzen oder zu nutzen (und das entsprechend auch tun). Es hat in der Vergangenheit auch niemanden gegeben, der gesagt habe „Lasst uns nun Wikis zum Wissensmanagement bei der Softwareentwicklung nutzen“. Gerade diese Entwicklungen sind für einen „kreativen Wildwuchs“ verantwortlich. Inzwischen werden Wikis jedoch von einigen Einheiten, zum Beispiel bei Sun Learning Services gezielt und qua-

---

<sup>3</sup> [blogs.sun.com](http://blogs.sun.com)

<sup>4</sup> [wikis.sun.com](http://wikis.sun.com)

<sup>5</sup> [www.atlassian.com/software/confluence](http://www.atlassian.com/software/confluence)

si „kontrolliert“ eingesetzt und erfreuen sich (auch in zentralisierter, strukturierter Form) einer großen Beliebtheit. Das zeigt sich auch am zentralen Wikiportal von Sun: Wikis.sun.com ist erst sein Mai 2007 online und enthält derzeit bereits 138 Wikis!

Die Vorteile der Wikis beschreibt Josef Holy folgendermaßen: Der größte Vorteil sei, dass die Informationen in den Wikis lebendig und dynamisch sind. Außerdem sind die Teilnahmebarrieren bei Wikis nur sehr klein (insbesondere in einer technologischen Firma wie Sun). Zudem arbeiten die Mitarbeiter aus seiner Sicht auch als und im Wiki-Netzwerk besser: Sie organisieren und strukturieren ihre Informationen eher „natürlich“, so dass sie für andere einfacher zugänglich und verständlich ist. Teams können ihre eigene Informations-Infrastruktur eines Wikis nach ihren eigenen Bedürfnissen aufsetzen.

Der Mehrwert, der sich durch die einfache Schaffung eines freien Wissensraumes eingestellt hat, hat aber auch seine Kehrseite: Ähnlich wie in der Online-Enzyklopädie Wikipedia ist nicht immer nachvollziehbar, welche Informationen für eine Problemstellung relevant sind und wie sie mit anderen Artikeln in Verbindung stehen. Auch die Validität der einzelnen Wiki Artikel ist nicht per se gegeben.

Eine zusätzliche Problematik besteht in den wachsenden Wissensinseln. Diese entstehen durch eine Vielzahl von Orten, an denen das Wissen gespeichert wird und entwickeln sich vor allem dann zu einem Problem, wenn die Beteiligten nicht auf dieselbe Wissensbasis Zugriff haben oder diese derart heterogen ist, dass Redundanzen unvermeidbar werden.

Der Unübersichtlichkeit, die sich zwangsläufig durch das individuelle Anlegen von Wikis ergibt, wird im Wiki-System von Confluence durch das Erstellen von individuellen RSS-Feeds entgegenwirkt. RSS-Feeds, also das Verteilen der Information über aktuelle Änderungen durch ein plattformunabhängiges XML-Format, werden von vielen Softwaretools angeboten (siehe dazu auch Kapitel 2.6). Es ermöglicht den Benutzern immer über Änderungen informiert zu sein, ohne dass die entsprechenden Seiten im Web besucht werden müssen. Bei Wiki-Systemen werden die RSS-Feeds grundsätzlich auf die History-Funktion angewandt: d.h. die Veränderungen zwischen zwei Versionen im Wiki werden durch den RSS-Feed in einem RSS-Reader lesbar.

Das System Confluence hat zusätzlich einen RSS-Composer integriert: Dieser Composer bietet den Benutzern die Möglichkeit, über alle Wiki-Bereiche hinweg nach Schlagworten, nach Titel oder Volltext zu suchen und die Suchanfragen als individuelle RSS-Feeds zusammenzustellen. Das bedeutet, dass nicht nur die letzten Änderungen eines einzelnen Wikis übermittelt werden, sondern dass ausgesuchte Informationen unterschiedlicher Seiten und Bereiche übertragen werden können.

### 3.3 Die kritischen Erfolgsfaktoren

Die Umsetzung von Wikis bei Sun zur Dokumentation von Wissens- und Software-Entwicklungsprozessen ist derzeit brauch- und handhabbar, auch wenn aus den oben geschilderten Gründen kaum konkrete Zahlen vorgelegt werden können. Informationen einfach und schnell zu finden funktioniert allerdings nur solange, wie die verschiedenen Wiki-Systeme (oder Wiki-Spaces) innerhalb einer Plattform laufen, auf die der RSS-Feed-Composer zugreifen kann. Da bei Sun auch weitere Wiki-Systeme genutzt werden, ist jedoch kein Zugriff auf die Gesamtheit der bei Sun verfügbaren Wiki-Systeme möglich.

Sun sieht sich durch die Freigabe des Quellcodes und die Mitarbeit weltweiter Benutzergruppen mit dem Phänomen konfrontiert, dass das Wissen über die Softwarebausteine nicht zentral entsteht und somit auch nicht zentral dokumentiert werden kann: Das Wissen über die aktuellen Softwareentwicklungen ist nicht nur in unterschiedlichen Dokumentationsdatenbanken, sondern in der gesamten Community verankert. Darüber stellt es auch eine Herausforderung dar, eine Plattform für die unterschiedlichen Formate bereitzustellen: Bei der Entwicklung von Softwareentwicklungsplattformen fallen neben der Dokumentation der Codeentwicklung auch Softwareanforderungen und Use-Cases, Codeabschnitte, Best-Practice-Beispiele, Patches, Bug-Reports und (multimediale) Tutorien an. All diese Informationen zentral verfügbar zu machen, stellt ein hohes Maß an Komplexität dar, die in den Anfängen durch unterschiedliche Plattformen abgebildet wurden.

Gerade in der Entwicklung von Open-Source-Projekten ist es üblich, dass die Kommunikation und Dokumentation mit Hilfe von Wikis durchgeführt wird: „Nahezu alle Open-Source-Projekte bedienen sich Wikis zu Planung von Funktionalitäten, Dokumentation und Interaktion mit den Benutzern. Vereinigungen von Open-Source-Projekten, wie Apache Jakarta<sup>6</sup> oder Codehaus<sup>7</sup> bieten den von ihnen gehosteten Projekten ein Wiki zur Software Entwicklung an“ [John et al. 2005]. Auch bei Sun wurde frühzeitig dazu übergegangen, Wikis für diese Dokumentationen einzusetzen. Gerade der Mehrwert von Wikis wird speziell bei NetBeans in Prag groß geschrieben. Neben dem frei zugänglichen, öffentlichen Wiki<sup>8</sup> sind weitere interne Wiki-Systeme im Einsatz, die die laufenden Aktivitäten der einzelnen Arbeitsteams unterstützen.

Mit dem Erreichen einer kritischen Masse an Artikeln innerhalb eines Wikis ist es notwendig, sich über Struktur, Navigation und vor allem über die Pflege und Aktualisierung der Artikel Gedanken zu machen. Die Online-Enzyklopädie Wikipedia beschäftigt zu diesem Zweck so genannte „Gärtner“, die sich mit der Problematik unsauberer Wikiseiten auseinandersetzen, diese bereinigen und deren Struktur vereinheitlichen.

Den rechtlichen Rahmen für das Veröffentlichen von Informationen bietet hier (ähnlich wie auf der Seite [blog.sun.com](http://blog.sun.com)) ein „Wiki-Policy“-Dokument, das den Umgang mit vertraulichen Informationen sowie eine Richtlinie für die Verwendung des firmeninternen Wikis bereitstellt.

### 3.4 Wenn Wikis an ihre Grenzen stoßen: geplante Weiterentwicklungen bei Sun

Die Problematik, mit der sich die Entwickler konfrontiert sehen, ist auf zwei unterschiedlichen Ebenen gelagert. Zum einen werden von Wiki-Systemen nicht alle Bereiche, die für die Dokumentation und Unterstützung bei der Softwareentwicklung erforderlich sind, unterstützt. Zu dieser Erkenntnis kommen auch KRANZ und MERZ, die Wikis als ungeeignet für die Entwicklung von Source-Code (Programmquelltext), Issue-Tracking (Sammeln von Feh-

---

<sup>6</sup> [www.apache.org](http://www.apache.org)

<sup>7</sup> [www.codehaus.org](http://www.codehaus.org)

<sup>8</sup> [wiki.netbeans.org](http://wiki.netbeans.org)

lerberichten) als auch ungeeignet als Diskussionsforum bezeichnen [vgl. Kranz und Merz 2005]. Zum anderen ist es derzeit nicht möglich, über die Vielzahl der unterschiedlichen Systeme zentrale Suchabfragen abzuschicken und somit die gesammelten Dokumentationen als Wissensbasis zu erschließen.

Derzeit werden für die verschiedenen Arbeitsbereiche/Wikispaces unter anderem die folgenden Dokumentations- und Kommunikationssysteme verwendet: Dokumenten-Repository mit Produktdefinitionen und Use-Cases, Spezifikationen, Softwaredokumentationen, Studien und Berichte sowie Bugreports (Fehlerberichte). Darüber hinaus werden E-Mail und Mailinglisten, Weblogs für die neuesten Entwicklungen, Code-Repositories, Bug-Tracking-Systeme, CVS-Systeme verwendet. Diese Arbeitsbereiche sind in den meisten Fällen nicht strikt trennbar und vermischen sich inhaltlich immer häufiger.

Da es jedem Mitarbeiter freisteht, über [wiki.sun.com](http://wiki.sun.com) oder [wiki.netbeans.org](http://wiki.netbeans.org) eigene Arbeitsbereiche anzulegen, sind einzelne Entwicklergruppen dazu übergegangen, die gesamte Dokumentation über Wikis zu bewerkstelligen und nicht die bereits im Einsatz befindlichen Werkzeuge (Forum, Mailinglisten, Dokumenten-Repositories etc.) zu verwenden.

Die Schwierigkeit, die sich aus dieser Entwicklung ergibt, liegt auf der Hand: Die Auffindbarkeit von Information in der gesamten Wissensbasis wird durch die unüberschaubare Verbindung der unterschiedlichen Systeme immer komplexer, der Mehrwert der sich ursprüngliche durch die Verwendung einer frei konfigurierbaren Wissensstruktur entwickelt hat, verliert zunehmend an Bedeutung, da die Informationen nicht mehr auffindbar werden.

Sun Microsystems versucht, diesem Problem mit zielgerichteten Strategien zu begegnen: Beispielsweise wird versucht, die bestehenden unterschiedlichen Systeme über eine Portallösung zugänglich zu machen, um so zumindest eine Volltextsuche über die verschiedenen Plattformen zu ermöglichen. Aufgrund der Heterogenität der Systeme führt die Volltextsuche jedoch selten zu einem brauchbaren Ergebnis.

Seit dem Jahre 2004 wurde daher zusätzlich mit semantischen Wikis experimentiert (siehe dazu auch Kapitel 5.4). Dabei handelt es sich um Systeme, bei denen es möglich ist, einzelne Artikel auf Basis einer bestehenden Wissensstruktur (Taxonomie, Ontologie) qualitativ zu annotieren. Diese Annotationen sind nicht nur von Maschinen lesbar, sondern auch für Menschen verständlich, wodurch vom System automatisch Beziehungen zwischen den Artikeln abgeleitet werden können. Man erhofft sich dadurch eine automatische Verknüpfung von Informationsbausteinen, die aus gleichen oder ähnlichen Wissens- und Arbeitsbereichen stammen, auch wenn sie in unterschiedlichen Systemen gespeichert sind.

Die ersten Systeme die es ermöglichten, semantische Annotationen zu gespeicherten Informationen hinzuzufügen, wurden von Sun in diversen Pilotprojekten erprobt, aus denen sich Ende 2005 eine eigene Arbeitsgruppe formierte, die die Analysen von semantischen Wikis für den Einsatz bei Sun erproben sollte. Derzeit sind zehn derartige Systeme implementiert, fünf davon befinden sich in einem Demonstrationsstatus.

## 4 Erkenntnisse über den Einsatz von Social Software in Unternehmen

Wie in dem vorliegenden Beispiel von Sun verdeutlicht wird, ist die Einführung von Social Software vor allem eine Frage der Unternehmenskultur und damit eine strategische Entscheidung, die vom Unternehmen nicht nur angeboten und unterstützt, sondern auch gelebt werden muss. Gleichzeitig zeigt das Beispiel Sun auch, wie es einem Unternehmen gelingen kann, ein „Enterprise 2.0“ zu werden, also z. B. erfolgreich organische Wissensentwicklung und Innovation durch Kombination realisieren kann (siehe oben und [Berlecon 2007]).

In Abb. 1 zeigen wir die Zusammenhänge zwischen einer allgemeinen Firmenphilosophie (im Sinne von Unternehmensleitlinien und der herrschenden Firmenkultur), der konkreten Strategie sowie des gewählten technologischen Zugangs auf und nennen dazu einige wichtige Beispiele für die Umsetzung bei Sun. Nur wenn alle drei Bereiche zueinander passen oder aufeinander abgestimmt werden können, kann der Schritt zum Enterprise 2.0 gelingen.

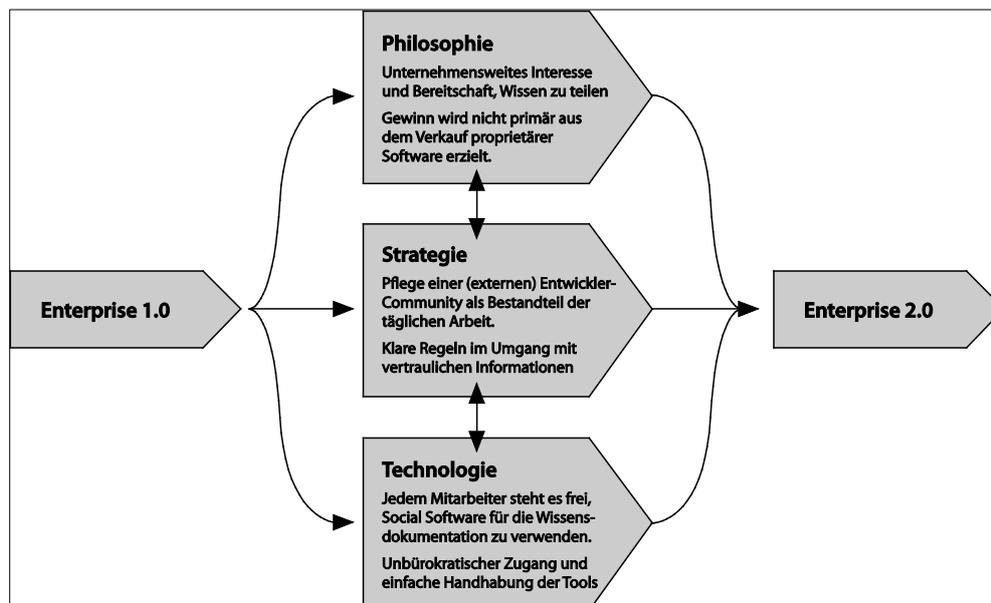


Abb. 1: Erfolgsfaktoren beim Wandel vom Unternehmen 1.0 zum Enterprise 2.0 am Beispiel Sun

### 4.1 Wandel der Unternehmens- und Kommunikationskultur

Der Einsatz von Social Software ist nicht nur eine Entscheidung für eine Technologie, sondern auch eine Entscheidung für eine bestimmte Unternehmens- und Kommunikationskultur. Die Implementierung von Social Software zeigt sich nur in solchen Projekten, Gruppen oder Firmen als effektiv und gewinnbringend, wenn eine Bereitschaft aller Beteiligten vorliegt, Informationen und Wissen (mitzu-)teilen und öffentlich zu dokumentieren. Darüber hinaus müssen die entsprechenden Rahmenbedingungen vorliegen, d.h. dass diese Bereitschaft vom Unternehmen und den Vorgesetzten auch honoriert und wertgeschätzt wird.

Als weiteren Erfolgsfaktor kann man die Zeit ansehen, die die Mitarbeiter der Pflege der Community (sowohl intern als auch extern) einräumen dürfen. Wissen zu teilen ist keine

Frage des Privatvergnügens und kann/soll daher nicht in der Freizeit stattfinden. Vielmehr verschmelzen die Prozesse der Wissensteilung/-generierung mit der Pflege der Community, sodass keine Grenze für diese Prozesse gezogen werden darf.

Die Freiheit und Offenheit im Umgang mit Wissen sollte aber nicht mit einer grenzenlosen Öffnung von Firmeninterna verwechselt werden. Selbstverständlich ist es notwendig, interne strategische Informationen auch intern zu halten und sich über den Umfang der veröffentlichten Informationen im Klaren zu sein. Das bedeutet für Unternehmen, dass sie diese Regelungen klar und nachvollziehbar den Mitarbeitern kommunizieren. Entsprechende Informationen sollten ebenso gut durchdacht, wie auch einfach zugänglich zur Verfügung gestellt werden.

## 4.2 Open Source – Open Thoughts, zwei ähnliche Modelle

Die ursprüngliche Intention der Open-Source-Bewegung hat es sich zum Ziel gesetzt, durch Teilen und den Austausch über veröffentlichte Software(-teile) das Wissen darüber zu vergrößern und damit den Erfolg der Software zu beschleunigen. Dieser Gedanke ist auch in der Social-Software-Community nicht anders, wo das gemeinsame Wissen durch das Teilen von (einzelnen) Ideen vergrößert werden kann. Dementsprechend könnte man Social Software als Netzwerk der „Open Thoughts“ bezeichnen.

Dass sich in dieser Diskussion der Gedanke des geistigen Diebstahls aufdrängt, ist nicht neu. Dieser Gedanke und die Auswirkungen auf eine freie Wissens-Community wurden im Artikel von Efimova, Fiedler, Verwijs und Boyd behandelt, der die positiven Aspekte dieses erlaubten Diebstahls („Legitimised Theft“, so der Titel des Artikels) beleuchtet [vgl. Efimova et al. 2004].

## 5 Ausblick: Zukünftige Entwicklungen bei Sun

Der bisherige Einsatz von Social Software bei Sun ist eine Erfolgsgeschichte: Wikis sind als Wissenmanagement-Tool in der Software-Entwicklung nicht mehr wegzudenken und haben so auch zum wirtschaftlichen Erfolg von Sun beigetragen. Trotzdem besteht, wie oben ausgeführt, für Sun dringender Handlungsbedarf um die Übersichtlichkeit der Wikis und damit den Nutzen zu erhalten.

Die Notwendigkeit, die heterogenen Systeme in ihrer Gesamtheit überschaubar zu gestalten, geht aus heutiger Sicht nicht ohne die Verwendung intelligenter und (semi-)automatischer Systeme. Die Informationsobjekte werden durch die intelligenten Verknüpfungen zu Wissensräumen zusammengefasst, die durch vordefinierte und maschinenverständliche Strukturen (Taxonomien, Ontologien) beschrieben werden können. Die Semantic-Web-Technologien stellen die dafür notwendige technische Infrastruktur zur Verfügung.

Wenn sich jedoch das Wissens zum einen in unterschiedlichen Datenbanksystemen und zum anderen in einem scheinbar unstrukturierbaren und komplexen Zustand befindet (wie das bei Social Software häufig der Fall ist), ist es derzeit nicht möglich, semantische Technologien für das Erfassen, Beschreiben und vor allem Finden von relevanten Informationen einzusetzen.

Das FP-7 Projekt „KIWI – Knowledge in a Wiki“<sup>9</sup> versucht, in den kommenden drei Jahren diese Probleme zu lösen und ein System zu entwickeln, das es Benutzern ermöglicht, über unterschiedliche Datensysteme und Formate hinweg auf qualitativ und semantisch beschriebene Inhalte zuzugreifen. Unter der Leitung von Salzburg Research wird konkret am Anwendungsfall Sun Microsystems/NetBeans in Prag (sowie einem zweiten Anwendungsfall) ein Systemframework entwickelt, das auf Basis des von Salzburg Research entwickelten semantischen IkeWikis<sup>10</sup> nicht nur klassische Wiki-Funktionalitäten zur Verfügung stellt, sondern auch eine Plattform für die Projektkollaboration und die (technische) Dokumentation von Softwareentwicklungsprozessen bietet.

Grundsätzlich prallen bei dieser Konzeption zwei Paradigmen aufeinander: Die flexiblen und von sozialen Prozessen geprägten Konventionen über die Zusammenhänge von Wissensbeiträgen in der Social Software auf der einen Seite und der Versuch einer Definition von Wissenszusammenhängen in Ontologien auf der anderen Seite. Zu zeigen, dass diese zwei Paradigmen aber nicht zwangsläufig einen Widerspruch bedeuten müssen, hat sich das KIWI-Projekt zum Ziel gesetzt: Es entwickelt ein Konzept, welches die Kombination von flexiblen und dynamischen Strukturen (Social Software) mit starren semantischen Vorgaben (Ontologien) ermöglichen soll. Durch diese Verknüpfung soll ein problemorientiertes Wissensmanagementsystem entstehen, das sowohl von Laien (der externen Community) als auch von Experten bedient und genutzt werden kann.

Die Firma Sun Microsystems hat ihre Projektpartnerschaft damit begründet, dass die vorherrschende Wissensdokumentation schon jetzt nicht mehr ausreichend ist und dringend neue Wege beschritten werden müssen, um in Zukunft konkurrenzfähig zu bleiben. Das Vertrauen in freie und flexible Strukturen (wie in Social Software) und die Überzeugung vom Mehrwert semantischer Technologien sind der Garant, dass Sun Microsystems mit dem KIWI-Projekt die Vorreiterrolle im Bereich der innovativen Unternehmensstrategien weiterhin innehalten wird.

---

<sup>9</sup> [www.kiwi-project.eu](http://www.kiwi-project.eu)

<sup>10</sup> [ikewiki.salzburgresearch.at](http://ikewiki.salzburgresearch.at)