

Einleitung

Knowledge Management (KM) – Wissensmanagement – wird den meisten Lesern ein Begriff sein und beschäftigt sich mit Wissensprozessen aller Art. Open Source ist ein ebenfalls oft gehörter Begriff und wird daher auch den allermeisten Lesern bekannt sein. Gleichzeitig ist es nötig hier kurz darauf einzugehen, wie und in welcher Art diese beiden Begriffe in diesem Buch verwendet werden, da es sich in beiden Fällen eigentlich um Überbegriffe und keine scharf umrissenen Gebiete handelt.

Zuerst ist es aber wichtig, sich grundsätzlich zu überlegen, was „**KM und Open Source**“ überhaupt heißen kann und welchen Sinn das Ganze überhaupt macht: KM hat in den vergangenen Jahren viele Hoffnungen enttäuscht. Trotz allem wird die prinzipielle Notwendigkeit, sich besser und effizienter um das Phänomen Wissen zu kümmern, nicht bezweifelt. Aktuell wird auch des Öfteren von Intelligence anstelle von Wissen gesprochen (bis zu einem gewissen Grad um den Begriff Management in den Hintergrund zu drängen).

Viele der schon umgesetzten **KM-Projekte** waren „**IT lastig**“ und haben trotz enorm hoher Investitionen nicht annähernd das gewünschte Ergebnis geliefert. Dieses Faktum nehmen viele Kritiker zum Anlass, um der IT die Schuld zu geben, wobei diese Argumentation nicht vollständig entkräftet werden kann – sie geht aber am Kern des Problems vorbei.

Prinzipiell betrachtet braucht man keine **IT**, um Knowledge Management umzusetzen. Ein KM-Projekt kann aus rein organisatorischen Maßnahmen bestehen und beachtliche Erfolge erzielen. Gleichzeitig werden in fast allen Firmen immer mehr Aktivitäten mithilfe des Computers erledigt. Daraus folgt, dass ein umfassend angelegtes KM-Projekt sich sehr wohl mit IT auseinandersetzen muss, um die wichtigen und notwendigen organisatorischen Veränderungen abbilden und unterstützen zu können. Daraus ergibt sich die Notwendigkeit bei fast allen KM-Projekten, zuerst die organisatorische Seite zu betrachten und danach sehr wohl auch die IT einzubeziehen.

Der Einsatz eines der vielen kommerziell angebotenen KM-Systeme ist aber eine sehr kostspielige Angelegenheit. Daher stellt sich die Frage, welche Bereiche von KM-Systemen man mit kostenlosen Produkten abdecken kann.

Da es bei KM-Projekten sehr stark um die Individualisierung und Anpassung geht, kommen eigentlich primär Open Source Produkte in Frage.

Man sieht also, dass sich das Thema „KM und Open Source“ ganz natürlich ergibt und eine sehr interessante Frage darstellt. Bevor es aber weiter um diesen Zusammenhang geht, ist es wichtig ein paar grundsätzliche Definitionen vorzustellen, die einerseits das Verständnis des Autors bezüglich der behandelten Themen zeigen und gleichzeitig eine Basis für den Rest des Buches sind.

1.1 Definitionen (Knowledge, Knowledge Management, Open Source, ...)

Wenn man Knowledge Management definieren will, stolpert man sofort über den Begriff: Knowledge – Wissen. Dieser Begriff ist leider (wie in Folge auch KM) nicht eindeutig definiert und es gibt auch kein gemeinsames Verständnis zwischen den verschiedenen KM-Richtungen über die Definition dieses Begriffs.

1.1.1 Definition von Knowledge

Daher bleibt nur übrig sich der Sache soweit als notwendig anzunähern und den Leser zu bitten, diese Sichtweise bei der weiteren Lektüre im Hinterkopf zu halten. Einer der relativ pragmatischen KM-Forscher (seit 1994 im KM Umfeld aktiv und einer der aktivsten Teilnehmer der KMCI Liste [26]) ist dabei Denham Grey und daher wird hier folgende Definition von ihm zitiert:

Context-relevant, validated information clusters that emerge when people somehow deal/interact with information elements/people.

Man sieht sofort, dass es wie auch in klassischen Definitionen von Wissen, um die überprüfte Information geht, wobei im Kontext von KM auch sehr stark auf die Kontextkomponente und die Interaktion von Personen mit Informationen und untereinander eingegangen wird. Im Kontext dieser Definition muss man noch kurz auf den Begriff Information eingehen (der ja wie Knowledge selbst sehr viele Bedeutungen hat – zum Beispiel den Informationsbegriff nach Shannon in der Informatik). Ohne auf die Debatten um diesen Begriff näher einzugehen, soll noch eine weitere Definition angeführt werden, die im KM-Umfeld gerne genommen wird und Information als: „data endowed with relevance and purpose“ sieht (siehe Drucker, P.E. (1995): „The Post Capitalistic Executive.“ in P.E. Drucker (ed.): Management in a Time of Great Change, New York: Penguin.).

Wenn man **KM-Literatur** liest, wird man viele andere Definitionen der Begriffe Information und Knowledge finden (für eine Übersicht bietet sich eine sehr gute Arbeit von Alavi und Leidner an [1]). Gleichzeitig gibt es dabei

viele Übereinstimmungen – Knowledge wird meistens als personalisierte (individualisierte) Information gesehen. Ein wichtiger Schluss aus dieser Definition ist, dass man Knowledge nicht elektronisch speichern kann.

Insgesamt möchte sich der Autor der Meinung vieler seiner Kollegen anschließen, dass es ganz sicher kein Knowledge darstellt, wenn man Unmengen an Information elektronisch gespeichert hat. Hier liegt auch die Erklärung warum rein „IT lastige“ Initiativen scheitern mussten und nur geringe Ergebnisse bei hohen Kosten geliefert haben (wenn man es genau betrachtet, gab es ähnliche Ansätze im eLearning und es ergaben sich ähnliche Probleme).

1.1.2 Definition von Knowledge Management

Da es keine klare Definition von Knowledge gibt, ist es sinnlos nach einer eindeutigen Definition von Knowledge Management zu suchen. Daher wird noch einmal Denham Gray zitiert, der Knowledge Management folgendermaßen charakterisiert:

Broad-based discipline: mine (analyze) data; capture, create, store, catalog, validate, transform, and disseminate/share information; capture, emerge, store, catalog, and disseminate/share knowledge.

Kurz gesagt, Knowledge Management umfasst alle Aktivitäten, die sich mit Daten, Informationen und Knowledge beschäftigen (was größtenteils auch von Informationsmanagement geleistet wird). Die Schwierigkeit der Abgrenzung von Knowledge Management und anderen Gebieten wird damit auch deutlich und man kann sagen: Knowledge Management ist alles oder nichts. Auf so einer Basis lässt sich natürlich nicht allzu viel feststellen. Daher soll noch auf Abschnitt 2.1 verwiesen werden, der sich mit konkreten Zielen von KM beschäftigt. Zusammen mit der groben Definition wird das im Weiteren als Basis für dieses Buch verwendet.

1.1.3 Das Verhältnis von Knowledge Management zu IT

Nach diesen Definitionen und Beschreibungen stellt sich die Frage, ob KM überhaupt direkt mit IT zusammenhängt. Wie schon kurz diskutiert, ist KM prinzipiell von IT unabhängig und man kann sich durchaus KM-Projekte vorstellen, die nur aus organisatorischen Maßnahmen bestehen. Je umfassender allerdings eine KM-Initiative wird und je stärker eine Organisation ihre Prozesse mit IT Systemen unterstützt, desto notwendiger ist eine Einbeziehung von IT.

Insgesamt ist eine umfassende KM-Initiative ohne IT Unterstützung kaum mehr denkbar. Gleichzeitig nimmt auch der Anteil von IT an allen Unternehmungen und Prozessen immer mehr zu. Es bleiben selbstverständlich noch genug Möglichkeiten, KM-Ansätze ohne IT einzubringen. Diese sind – wie auch der Titel zeigt – nicht Gegenstand dieses Buches (und es gibt eine große Menge an guten und passenden Büchern zu diesem Thema).

1.1.4 Definition von Open Source

Obwohl Open Source an sich ein Begriff der IT ist, hat auch er keine eindeutige Definition, sondern stellt eine Art Oberbegriff dar. Generell geht es darum, dass der Quellcode (Source) eines Produkts zugänglich ist. Bei den „Details“ gibt es aber deutliche Unterschiede.

Prinzipiell kann der **Quellcode** von Software **zugänglich** sein, ohne dass man etwas damit tun dürfte (außer natürlich ansehen). Die meisten Vertreter von Open Source versuchen den Begriff aber nur auf Software anzuwenden, bei denen eine Veränderung und Weitergabe des veränderten Source Codes möglich ist. Im Weiteren wird primär auf solche Software eingegangen und explizit darauf hingewiesen, wenn es sich um Software handeln sollte, die ohne Source Code verfügbar ist beziehungsweise bei der man den Source Code nicht ändern darf.

Wenn man sich also die **engere Definition** von Open Source ansieht, dann geht es darum, die Modifizierbarkeit des gelieferten Sources zu garantieren. Es gibt dazu eine entsprechend benannte Initiative namens OSI (Open Source Initiative: <http://www.opensource.org/>). Auf der entsprechenden Homepage findet man eine Menge von Informationen über Open Source und eine Auflistung aller relevanten Lizenzen, die die Kriterien erfüllen.

Neben dem Begriff **Open Source** gibt es auch noch den eng verwandten Begriff **Free Software**. Wie auch bei Open Source, gibt es Verwechslungsgefahr (in diesem Fall mit Software, die kostenlos ist, aber keinen Source Code mitbringt oder dieser nicht verändert werden darf). Für dieses Buch ist die Kategorie von kostenloser Software ohne Quellcode nicht interessant (weil bei KM meist zu hoher Anpassungsbedarf besteht, als dass man ohne den Source Code auskommen könnte). Der „Evangelist“ der für dieses Buch interessanten Free Software Definition ist Richard Stallman mit seiner Free Software Foundation. Auf ihrer Homepage (<http://www.fsf.org>) findet man neben einer Menge Hintergrundinformation auch die beiden wichtigsten Open Source/Free Software Lizenzen, die GPL (General Public License) und die LGPL (Lesser General Public License). Wenn man sich die Statistiken auf einer der größten Open Source Sammlungen ansieht (<http://freshmeat.net/stats/>) ist auf den ersten Blick offensichtlich, dass die GPL aktuell dominiert. Besonders interessant für manchen Leser mag vielleicht noch der relativ hohe Anteil der Projekte mit BSD (Berkeley Software Distribution) Lizenz sein (zum Zeitpunkt des Verfassens dieses Buches ca. 5%). Bei dieser Lizenz kann man eine veränderte Version kommerziell vertreiben. Bei der zum Zeitpunkt der Drucklegung größten Open Source Sammlung (<http://sourceforge.net>) sind ca. 96% der Projekte mit einer „OSI Approved“ License (also wo man wirklich von Open Source sprechen kann). Davon wiederum haben ca. 7% die BSD Lizenz (die meisten unterliegen der GPL/LGPL). Allerdings hatten nur 2/3 der Projekte eine eindeutig zuordenbare Lizenz. Eine nähere Betrachtung von Open Source im Allgemeinen und auch der Lizenzthematik gerade auch bei zum Teil kommerzieller Verwendung findet sich zum Beispiel im Buch „Open-Source-

Software“ von Heinz Sauerburger [52] oder einem Werk mit gleichem Titel von Bernd Brügge, Dietmar Harhoff und Arnold Picot [7].

Insgesamt ist Open Source vor allem in zweierlei Hinsicht für eine KM-Implementierung interessant:

1. Es gibt keine (hohen) Lizenzkosten.
2. Man kann die Software beliebig ändern.

Aus Sicht der Open Source Vertreter kommt noch dazu, dass der vorhandene Source Code erweitert und verbessert wird und man damit der Gemeinschaft auch wieder zurückgibt (anstatt nur zu nehmen). Deshalb wird hier die Kurzfassung der Definition von Open Source zitiert (von <http://www.opensource.org>):

Open source promotes software reliability and quality by supporting independent peer review and rapid evolution of source code.

Besonders interessant wird KM durch die Tatsache, dass es kaum mehr einen Bereich der IT ohne gute Produkte gibt. In manchen Nischen gibt es sicher noch Defizite und bei Neuentwicklungen kann es manchmal etwas dauern, aber generell gibt es fast alles in einer Open Source Variante und der „Abstand“ zu den kommerziellen Systemen wird in jeder Hinsicht geringer (aktuell sind zum Teil noch mehr Funktionen vorhanden). Auch benötigen KM Systeme überwiegend grundlegende oder breit eingesetzte (Mainstream-) Technologien, sodass es hier kaum nennenswerte Schwierigkeiten gibt. Das ist das vielleicht schlagendste Argument, warum das vorliegende Buch Sinn macht.

1.1.5 Häufige Fragen bei Open Source

Im Zusammenhang mit Open Source/Free Software gibt es eine Reihe von Unterschieden zu konventioneller, kommerzieller Software. Das führt einerseits zu Unsicherheiten bei potenziellen Anwendern und bedeutet andererseits, dass man auf bestimmte (andere) Dinge acht geben muss. Die wichtigsten Aspekte sind dabei folgende:

Unterstützung/„Support“ durch den Hersteller: Die ist bei Open Source in anderer Form und in sehr unterschiedlicher Qualität gegeben (wie ja auch bei kommerzieller Software). In vielen Fällen gibt es sehr aktive „Communities“ und bemühte Entwickler zu einem Produkt. In anderen Fällen gibt es vielleicht nur sehr wenige Personen mit dem notwendigen Know-how und es wird versucht, die Kosten der Entwicklung mit kostenpflichtigem Support abzudecken (etwa bei dem Application Server JBOSS). Daraus ergibt sich die Schlussfolgerung, dass man auch bei Open Source für kritische EDV Anwendungen für die notwendige Unterstützung etwas bezahlen muss. Die Freiheit, die man hat, besteht darin, den besten und kostengünstigsten Dienstleister zu beauftragen. Vor allem kann das selbst

bei einer Vielzahl von Produkten ein einziger externer Partner mit relativ geringen Kosten sein. Jedenfalls muss man dieses Thema auch bei einem Open Source System diskutieren und eine Lösung finden.

Fehlende Weiterentwicklung: Ein relativ beliebtes Argument gegen den Einsatz von Open Source ist das Risiko, dass eine Software nicht mehr gewartet oder weiterentwickelt wird. „Vergessen“ wird darauf hinzuweisen, dass so etwas bei kommerzieller Software genauso vorkommt und dort die Auswirkungen viel dramatischer sind, weil kein Dritter einspringen kann (da kein Source Code und keine Rechte vorhanden). Das Risiko, dass so etwas passiert, ist bei Open Source Applikationen vielleicht etwas höher als bei einer Firma. Gleichzeitig reden wir hier von einem quantitativen und keinem qualitativen Unterschied. Wenn man sich generell gegen dieses Risiko absichern will, muss man sich für eine breit eingesetzte Lösung entscheiden. Nur so ist „garantiert“, dass ein Produkt bei Problemen von anderer Seite weitergeführt wird. Das gilt genauso für kommerzielle wie für Open Source Software.

Rechtliche Aspekte: Ebenfalls gerne vorgebracht wird die fehlende Rechtssicherheit bei Open Source. Eine fundierte Darstellung und Analyse dieser Thematik kann leider nur von Experten vorgenommen werden. Generell ist diese Thematik in Europa weniger kritisch als in den USA, weil es noch keine Software Patente gibt. Weiters sind viele der angeführten Probleme nicht Open Source spezifisch. Auch eine kommerzielle Software kann diverse Rechte Dritter verletzen und prinzipiell kann auch hier ein Anwender verklagt und verurteilt werden. Hier wird sich allerdings meist die Herstellerfirma damit befassen (müssen) und sich auch darum kümmern, solange es sie noch gibt. Eine Unsicherheit bleibt hier aber auf jeden Fall, da es noch praktisch keine Präzedenzfälle gibt. Damit wird aber auch aufgezeigt, dass es relativ unwahrscheinlich ist, überhaupt verklagt zu werden (unabhängig von der Frage eines Verschuldens). Damit ist das Risiko eines internen Open Source Einsatzes zumindest relativ gering. Trotzdem sollte man Augen und Ohren offen halten, um neue Erkenntnisse oder Veränderungen in der zugrunde liegenden Situation mitzubekommen. Seit ca. 2005 gibt es zu dieser Thematik auch Bücher im deutschen Sprachraum. So wird etwa im Text „Die GPL kommentiert und erklärt“ des Instituts für Rechtsfragen der Freien und Open Source Software [21] auf die am häufigsten eingesetzte Open Source Lizenz näher eingegangen. Eine allgemeinere Darstellung findet man bei Spindler [57]. Im englischen Sprachraum gibt es dagegen wesentlich mehr Texte, die zum Teil auch schon deutlich länger erhältlich sind. Insbesondere, wenn man sich für die Unterschiede der verschiedenen Lizenzen interessiert, ist das Werk von Laurent [30] interessant.

1.2 Verwandte Gebiete

Wenn man sich KM ansieht, dann gibt es eine Menge von verwandten Gebieten und Feldern. Wie ja schon aufgezeigt wurde, gibt es keine klare Definition von KM und damit natürlich auch keine klare Abgrenzung zu anderen Feldern. Das macht verständlicherweise Schwierigkeiten bei der Zuordnung von einzelnen Methoden, Software, etc., ist aber eigentlich kein großes Problem, solange es um das Erreichen von definierten Zielen geht.

Das bedeutet vor allem, dass es praktisch unmöglich ist, einzelne Anwendungsfelder oder Produkte eindeutig zuzuordnen. Generell basiert dieses Buch auf einer eher weiteren Auslegung des Begriffs KM und vieles wird man auch unter einem anderen Namen „verkauft“ bekommen.

Insbesondere trifft dies auf die Begriffe **Informationsmanagement** und **Groupware** im Kontext von IT zu. Dabei wird Informationsmanagement oft (aber keineswegs ausschließlich) als Überbegriff für „Management Information System (MIS)“, „Decision Support System (DSS)“ oder „Expert Systems (ES)“ verwendet. Groupware auf der anderen Seite versucht das Zusammenarbeiten von Gruppen zu unterstützen und stellt meist eine Kommunikationsplattform aufbauend auf E-Mail, geteiltem Kalender und einem globalen Adressbuch dar.

Wenn man sich die Definition und Zielsetzung von KM ansieht, erkennt man sofort, dass diese Softwarekategorien wichtige Teile für eine KM Lösung darstellen. Es geht sogar noch weiter: Eine KM Initiative kann häufig mit einer der genannten Lösungen auskommen. Dann kann man diese unterstützende Software natürlich KM Tool oder KM System nennen, während sie gar nicht als solche benannt wurde (genauso gut kann es auch umgekehrt sein – nicht jedes Produkt, das sich heute KM System nennt, ist besonders gut geeignet, um KM Initiativen zu unterstützen, während es andere Aufgabenstellungen sehr gut bewältigt).

Unter anderen **verwandten Systemkategorien** sind zwei besonders wichtig. Einerseits Unternehmenssoftware (**Enterprise Resource Planning – ERP** – ist am prominentesten; Business Intelligence ein aktueller Begriff in diesem Umfeld) wie sie von SAP, PeopleWare (der Hersteller gehört inzwischen zu Oracle) oder den entsprechenden Oracle Produkten geliefert wird. Diese Kategorie beschäftigt sich mit der Unterstützung von insbesondere administrativen Aspekten (Inventar, Personal, ...). Die Produkte sind in der Zwischenzeit sehr mächtig und teilweise auch in Richtung KM erweitert worden (es gibt etwa ein entsprechendes Produkt von SAP). Aus einer Open Source Sicht kommen diese Systeme nicht als integraler Bestandteil in Frage (sehr wohl muss man sich aber um die Datenintegration kümmern). Daher wird im Weiteren auf Open Source Alternativen (zum Beispiel Compiere) eingegangen werden.

Die zweite relevante Kategorie ist **Customer Relationship Management (CRM)**. Diese Produktgruppe ist wesentlich jünger als ERP und dementsprechend sind der Markt und auch die angebotenen Produkte sehr

uneinheitlich. Teilweise wurden oder werden diese Produkte in Richtung KM erweitert. In jedem Fall sind sie wichtige Datenquellen, die es ein KM System zu integrieren sind. Natürlich gilt auch hier wieder, dass sich dieses Buch auf die Open Source Alternativen konzentriert (wie openCRX, SugarCRM oder Compiere).

Weiters wird noch auf eine wichtige Grundlagentechnologie eingegangen, nämlich **Künstliche Intelligenz (KI)** oder Artificial Intelligence (AI). Diese Technologie ermöglicht es Computern, nicht offensichtliche Zusammenhänge aufzudecken und generell „schlau“ auszusehen. Von wirklicher Intelligenz kann beim aktuellen Stand der Technik aber leider keine Rede sein und die meisten Forscher halten das Ziel auch für nicht bald erreichbar (wenn überhaupt). Wichtig ist diese Technologie deshalb, weil sie den Hintergrund von vielen besonders KM relevanten Funktionen darstellt (und konsequenterweise auch entsprechend in den schon angesprochenen Produktgruppen eingesetzt wird).

Abschließend soll noch kurz angesprochen werden, dass es natürlich eine Vielzahl von weiteren relevanten und teilweise verwandten Gebieten und Bereichen der IT gibt. Insbesondere Taxonomien, Ontologien, Topic Maps oder das „Semantic Web“ als „Ablöse“ des heutigen World Wide Web zeigen interessante und wichtige Konzepte auf (alle drei Begriffe finden sich im Glossar). Soweit diese schon unmittelbar Sinn machen oder man sie zumindest berücksichtigen sollte, geschieht dies im weiteren Text.

1.3 Der Fokus des Buches

Nachdem der grobe Aufbau und Inhalt dieses Buches schon im Vorwort und Leitfaden präsentiert wurden, geht es an dieser Stelle hauptsächlich um eine **Konkretisierung des Fokus**. Ganz wichtig ist dabei der Bogen von den Grundlagen und der Vision von KM zu einem potenziell umfassenden KM System. Genau in diesem Zusammenhang müssen die Beschreibungen von Systemen, grundsätzlichen technischen Entscheidungen und Fragen der Einführung/Umsetzung verstanden werden.

In diesem Zusammenhang soll noch einmal auf die Wichtigkeit von **Kapitel 2** verwiesen werden. Dabei ist der Schwerpunkt einerseits eine Basis für eher technisch orientierte Leser zu bieten und andererseits das Verständnis des Autors von KM und KM Systemen zu präsentieren. Für einen gesamthafte Ansatz und angesichts der verschiedenen Ansichten zu KM ist das einfach eine Voraussetzung, die aber gleichzeitig möglichst kompakt gehalten ist und nicht „zu“ stark auf die Theorie eingeht.

Im **engeren Sinne** könnte man sagen, dass das Ziel dieses Buches ist, eine mittelfristige Entscheidung für das KM System einer kleineren oder mittelgroßen Organisation zu ermöglichen. Damit ist nicht gemeint, dass nicht auch größere Organisationen den Ansatz analog verwenden oder gar direkt umsetzen können. Allerdings kann die oft vorhandene organisatorische und

technische Komplexität im Rahmen dieses Buches nicht detailliert genug behandelt werden.

Beim zweiten Aspekt, der fundierten **technischen Entscheidung**, stellt das Buch einerseits entsprechende Technologien und Open Source Systeme vor und andererseits wird analysiert, wie eine schrittweise Einführung funktionieren kann und was das bedeutet. In diesem Zusammenhang wird natürlich auf die ständige Weiterentwicklung hingewiesen, die gerade im Open Source Umfeld sehr schnell vonstatten gehen kann. Gleichzeitig werden aktuelle Trends und Ausbaupläne im Text schon mitberücksichtigt. Auf dieser Basis ist eine tragfähige mittelfristige Entscheidung möglich und genau darauf arbeitet dieser Text hin. Das ist auch ein wichtiger Grund für die Präsentation einer Reihe von Erweiterungsmöglichkeiten, die nur selten unmittelbar im Fokus einer KM Initiative sein werden.

Der **gesamthafte Blickwinkel** wiederum ist vor allem für die organisatorischen Fragen der Einführung und Erweiterung eines KM Systems notwendig. Nur auf dieser Basis kann eine KM Initiative erfolgreich sein und die Wettbewerbsfähigkeit der Organisation entscheidend verbessern. Darüber hinaus zeigt der Ausblick des generellen Feldes KM und relevanter Technologien, dass nur eine Integration der Anstrengungen auf vielen Bereichen wesentliche Verbesserungen ermöglichen wird.



<http://www.springer.com/978-3-540-33076-9>

Open Source Knowledge Management

Hüttenegger, G.

2006, XIV, 285 S., Hardcover

ISBN: 978-3-540-33076-9