COST-BENEFIT ANALYSIS FOR THE DESIGN OF PERSONAL KNOWLEDGE MANAGEMENT SYSTEMS

Max Völkel and Andreas Abecker
FZI – Forschungszentrum Informatik Karlsruhe, Haid-und-Neu-Straße 10-14, Karlsruhe, Germany
voelkel@fzi.de,abecker@fzi.de

Keywords: Knowledge management, personal knowledge, cost-benefit analysis.

Abstract: Knowledge Management (KM) tools have become an established part of Enterprise Information Systems in the recent years. While traditional KM initiatives typically address knowledge exchange within project teams, communities of practice, within a whole enterprise, or even within the extended enterprise (customer knowledge management, KM in the supply chain, ...), the relatively new area of Personal Knowledge Management (PKM) investigates how knowledge workers can enhance their productivity by better encoding, accessing, and reusing their personal knowledge. In this paper, we present a cost-benefit analysis of PKM – where benefit comes from efficiently finding task-specific, useful knowledge items, and costs come from search efforts as well as externalisation and (re-)structuring efforts for the personal knowledge base.

1 INTRODUCTION

During the last decade, *Knowledge Management* (KM) has reached a consolidated status as a management discipline and has provided manifold impressing business success stories in an economic world that becomes increasingly knowledge-based. In balanced socio-technical solutions, which holistically combine organizational, psychological, and IT measures, *Enterprise Information Systems* often play a central, enabling role. Originally KM functionalities enter the area of standard EIS solutions (e. g. groupware functionalities in Microsoft Sharepoint), and, vice versa, sophisticated KM tool suites are stepwisely further developed towards a central *Information Backbone* of the enterprise (Maier, 2004).

However, in spite of all such developments, the most important area of knowledge creation and processing is still the most underdeveloped area of KM: namely that of the individual knowledge workers *personal* knowledge management (PKM) (Davenport, 2005). In the European Integrated Research Project NEPOMUK[1], we develop methods and tools for the *Semantic Desktop*, an integrative infrastructure for organizing, interlinking, querying and exploiting personal information from different everyday applications through common semantics-based metadata. Later on, such personally created, collected, and managed information shall – partly – be made available to the peer group of the end user (their team colleagues, project partners, buddy networks, ...) through peer-to-peer technology, thus fostering an integrated community-knowledge space.

As one of the first and most important, yet still largely unsolved, questions arises that of appropriate methods and tools to support (and seduce) the user to articulate more, and more complex, parts of their expert knowledge in computer-based forms – for storing, sharing, further developing it, etc. Obviously, the more structure the user formally represents upfront (e. g. by indexing or tagging content, by filling a given document template, by adding metadata, etc.), the better prepared is such personal knowledge for later efficient finding again, for context-specific reuse, or also for sharing with other users. But this is a central KM barrier: Chronically overloaded managers, technicians, and creative workers simply do not invest time and effort in extra work to prepare for far-off, potential KM benefits. Hence, assuming that knowledge workers act – at least to a significant extent – rationally and economically, it is a central question to better understand the trade-offs between actual

[1] `http:\\nepomuk.semanticdesktop.org` Part of this work has been funded by the European Commission in the context of the IST NEPOMUK IP - The Social Semantic Desktop, FP6-027705. Part of this work has been done in *WAVES – Wissensaustausch bei der verteilten Entwicklung von Software*, funded by BMBF, Germany.

costs and expected benefits of knowledge articulation and of adding more structure to articulated knowledge. Insights in this area are required for building suitable Personal KM systems which will be accepted and used by knowledge workers and which can prove their Return-on-Investment in an enterprise setting.

Outline. We first describe the relations between knowledge management (KM), personal information management (PIM), and personal knowledge management (PKM). In brief, PKM can be seen as the KM perspective on PIM or the personal perspective on KM. Then we look from a high-level on costs and benefits in PKM. We develop a unified knowledge model (UKM) and explain how it can represent documents and ontologies in a unified way. Using the UKM, we refine the cost model and explain general cost drivers and benefits of PKM. Finally we apply the resulting formula as an example to the possible extend on a semantic wiki.

1.1 Knowledge Management

Drucker (1985) was among the first to use the term *knowledge worker* when stating

> The most important contribution of management in the 20th century was to increase manual worker productivity fifty-fold. The most important contribution of management in the 21st century will be to increase knowledge worker productivity – hopefully by the same percentage. [...] The methods, however, are totally different from those that increased the productivity of manual workers.

At times when knowledge management was becoming popular, Nonaka and Takeuchi (1995) published a book on knowledge processes which describes two basic states of knowledge: tacit (implicit) and external. Later works (Despres and Chauvel, 2000) conclude that external and internal knowledge are two extremes on a spectrum, but do not exist in reality. Maurer (1999) states that knowledge resides in the heads of people and the computer can only store "computerized knowledge" which is to be understood as “shadow knowledge”, a “weakish image” of the real knowledge. The high-level processes in knowledge management are externalisation, internalisation, combination and socialisation (Nonaka and Takeuchi, 1995). North (2007) defines knowledge work as “work based on knowledge with an immaterial result; value creation is based on processing, generating and communicating knowledge.[2]”

[2]translation by the author

1.2 Personal Knowledge Management

The knowledge-based organisation is no more effective than the sum of its knowledge workers’ effectiveness (Davenport, 2005). Knowledge can be *embrained* (conceptual, implicit), *embodied* (tacit, implicit), *encultured* (shared beliefs), *embedded* (in processes) or *encoded* (symbolic, external) (Blackler, 1995).

We use the term *personal knowledge management* (PKM) to denote the *process of the individual to manage knowledge*. In contrast to general *knowledge management*, PKM denotes the perspective of the individual. This perspective is potentially better suited to explain individual motivations and behaviour, even in organisational contexts.

The term *personal knowledge* has already been used in (Polanyi, 1958), the term PKM appears in (Frand and Hixon, 1999; Mitchell, 2005). The fields “ePortfolio” and “personal learning environment” deal with similar topics.

Higgison (2005) defines personal knowledge management as “managing and supporting personal knowledge and information so that it is accessible, meaningful and valuable to the individual; maintaining networks, contacts and communities; making life easier and more enjoyable; and exploiting personal capital”.

> Organizing information is a central part of the inquiry process focused on making the connections necessary to link pieces of information. Techniques for organizing information help the inquirer to overcome some of the limitations of the human information processing system. In some ways the key challenge in organizing information is for the inquirer to make the information his or her own through the use of ordering and connecting principles that relate new information to old information. . . . (Avery et al., 2001)

The related topic of *Personal information management* (PIM) got academic attention around 2005 when the first PIM workshop was held. The report of the second workshop (Jones and Bruce, 2005) defines the term Personal Space Of Information (PSI) as the space that “ includes all the information items that are, at least nominally, under that person’s control (but not necessarily exclusively so).” This includes a number of analog and digital storage locations.

There is no sharp distinction between PIM and PKM, rather a difference of scope and perspective. PIM focuses on managing all the *information* around an individual, i. e. only encoded knowledge. PKM deals with embrained, embodied and encoded knowl-

edge, i. e. mostly with personal, self-authored artefacts. This paper tackles the continuum between embrained and encoded knowledge, with regards to costs and benefits.

Jones et al. (2001) introduces the problem of "keeping found things found" which reports on the difference between *knowing* something and merely *storing information*. E. g. obviously an unread email can be managed in the sense of information management, but the knowledge one can get from the email has not yet been realised.

The basic processes in PIM are (Jones and Bruce, 2005): 1. keeping (input of information into a PSI); 2. finding/re-finding (output of information from a PSI); and 3. meta-activities like mapping between information and need, maintenance and organisation . The role of handling an external memory is apparent. In the next section we analyse costs associated to each phase. Note that the basic processes are the same for PIM and PKM, the difference is the perspective. PKM emphasises information from the users mind encoded by the user himself. PIM has a broader focus. PKM focuses on the knowledge in the users mind evoked by a piece of information presented by the system.

Many real-world notes seem to fall *between* embrained and encoded (Blackler, 1995) or between implicit and external (Nonaka and Takeuchi, 1995). E. g. a personal note with the content "milk" is not really an external, encoded representation of the knowledge "I need to buy some milk today". On the other hand, without the note, the action to buy milk might not happen. Different audiences and topics require different degrees of explicitness (Boettger, 2005).

Definition. A *knowledge cue* is an artifact which evokes (acts as a cue to) some kind of knowledge in the readers mind. I. e. knowledge is either reactivated by consuming the artifact representing the knowledge cue or new knowledge is obtained by consuming the information, e. g. when reading a book. The reader and author of a knowledge cue are the same person; other persons might not be able to gain knowledge from using the artifact representing the knowledge cue.

It is not possible to define precisely the amount of knowledge contained in one knowledge cue. Roughly, one knowledge cue evokes one concept or family of related concepts. As an example, a shopping list consisting of n items contains n knowledge cues. A description of a single business innovation could be treated as one cue. Bernstein et al. (2007) uses the term "information scrap" to denote one or several knowledge cues. On aspect of PKM is efficient and effective management of knowledge cues.

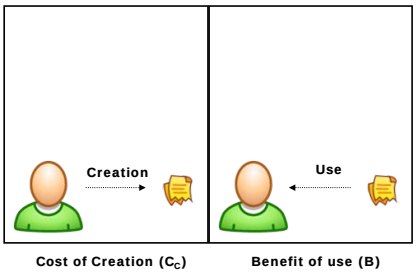

Figure 1: A Simplified Model for Cost/Benefit-Analysis in PKM without external memory stores.

2 COSTS AND BENEFITS IN PKM

We need to think in terms of investment, allocation of costs and benefits between the organizer and retriever (Glushko, 2006)[p. 24-31]. In PKM, organizer and retriever are the same person. Different from an organisational context, there is a personal motivation to organize knowledge.

Jones and Bruce (2005) reports on difficulties of evaluating PIM systems, as almost no person is willing to try out an experimental, potential unstable PIM system for a long period of time (months or years) for critical personal tasks such as remembering appointments, managing email or keeping personal notes. The analysis in this paper can help to evaluate systems used for PKM, e. g. also PIM systems.

This section analyses costs and benefits of an individual doing PKM from the perspective of a neutral observer. First we describe the PKM process without external storage and then extend this model.

2.1 Knowledge Creation and Use

If no external tools are used, the PKM process (c. f. Fig. 1) consists of

- *Knowledge creation* at a cost C_C and
- *Knowledge use* with a benefit B.

The **cost of knowledge creation** C_C is the amount of time, money and effort an individual spends on thinking, researching, experimenting, and learning. Cost can maybe best be measured by the amount of time spend – but how to measure the value? Maybe the time needed to re-create the knowledge? But if it takes long to re-create some knowledge that has a low benefit, then one would intuitively not assign a high value to it. Also knowledge might be cheap to create and store now but much more costly to re-create later, e.g. *Where have I been on a given day in 1999?*

The **value of knowledge** does not exist as such (Iske and Boekhoff, 2002); it depends highly on the

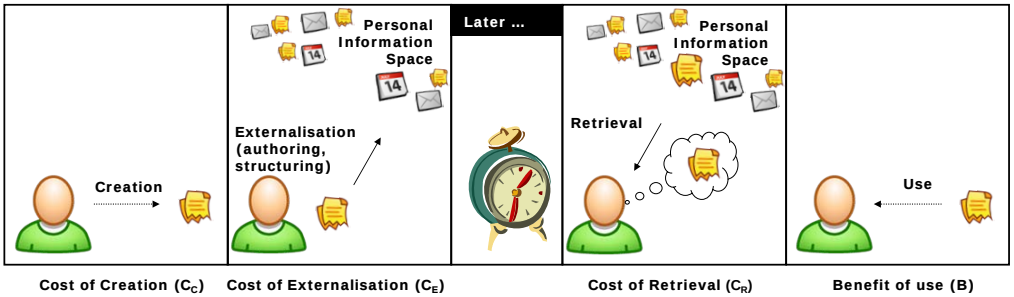

Figure 2: A Simplified Model for Cost/Benefit-Analysis in PKM. Goal: more benefit than costs ($B > C_E + C_R$).

task. The value of some knowledge can be defined as "increment in expected utility resulting from an improved choice" made possible by this knowledge (Varian, 1999). I. e. one estimates the value of the state of the world that would result from actions performed in absence of the knowledge (V_1) and compares it with the value of the state of the world resulting from actions taking in presence of the knowledge (V_2). Then the benefit B of having this knowledge for this task is the difference in value, i. e. $B = V_2 - V_1$. B then represents the additional created value or saved costs. B can (in theory) be measured in money, saved time, improved quality or better emotions. In practice, measuring the value of the state of the world is often hard to quantify, i. e. consider long-term effects and the difficulty to measure quality of better emotions. Most approaches resort to measure only costs (Feldman et al., 2005). Economists use an abstract "utility value" without defining a unit of measurement.

For comparing different PKM systems one needs a pragmatic way to take the benefit into account. We believe in PKM, the value of a certain piece of knowledge with respect to a task can be estimated by individuals on a Likert scale (Likert, 1932). The added value of using a knowledge management system for a given period of time (t) is the overall cost-benefit gain G which can be approximated by summing up all benefits B and subtracting the sum of all costs C ($G = B - C$). For a given amount of knowledge one can compare the costs of different PKM systems.

2.2 Using External Storage

In this section we extend the simple model introduced in Sec. 2.1 with an external storage system. The process of managing knowledge cues can be represented as (c. f. Fig. 2):

1. *Creation:* Knowledge is created at some costs C_C.
2. *Externalisation:* Some parts of the implicit knowledge become external. Knowledge cues are created. These user has externalisation costs C_E.
3. Time passes by and the author might start to forget some or all details of the articulated knowledge. Sometimes even the knowledge to know something is forgotten as well.
4. *Retrieval:* At a certain moment, while performing a certain task, the user initiates a retrieval process in his PKM system. As information retrieval system have become faster, the classic information retrieval measure of "time to execute query" becomes less relevant to determine the costs perceived by the user. The human-computer interaction becomes more often the bottleneck and cost-driver of efficient knowledge work.

   After having executed a query or performed a browsing step, the user reads the search results, and refines the search query. After some steps, the user either found one or several matching knowledge cues or cancels the search with no result. The process of reading through a list of search results takes time and therefore adds to the search costs.

   If a knowledge artifact is long in size, the time to read through it takes longer. If the desired knowledge is only a part of the artifact, reading through the artifact is thus additional search cost. All these costs are subsumed under retrieval costs C_R.
5. *Usage:* If results where found, the user has some benefit from having available the external knowledge or from remembering knowledge from the knowledge cue.

Esser (1998) analyses factors that determine when and which external memory humans use. Three variables were observed: Expected likelihood of successful remembering a piece of knowledge when stored in an external store, cost of storing it there and most importantly: perceived value of the knowledge to be stored. The higher the perceived importance of remembering the knowledge, the higher costs for storage were accepted.

The basic hopes of each person doing PKM should be:

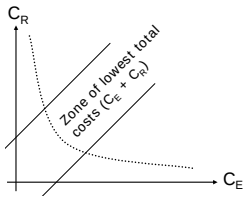

Figure 3: Assumed relation between externalisation costs (C_E) and retrieval costs (C_R).

- There is more benefit of having the knowledge (again) than it had cost to manage it, i. e. $C_E + C_R < B$
- Managing knowledge externally is cheaper than re-creating it from scratch, i. e. $C_E + C_R < C_C$.

Estimating the benefit of storing an item for later use ($C_E + C_R$)) compared with the expected costs to regenerate the contained knowledge later (C_C) is certainly hard. An assumption underlying this paper is that better organisation, structuring, and formalisation of content is expected to lower the costs of retrieval (Glushko, 2006).

2.3 Knowledge Representation

For a long time in history, knowledge cues were only stored in *documents*, first analog then digital. Digital document retrieval is the prime application domain of information retrieval (IR) techniques. The field of IR has a history of quantitative research, mostly focusing on *precision* and *recall*. These measures were first defined in Cleverdon et al. (1966). However, Cleverdon et al. (1966) proposed to measure not only these two factors but also "the extent to which the system includes relevant matter", and "the effort involved on the part of the user in obtaining answers to his search requests". Maybe because these two factors cannot so easily be measured automatically, they were mostly ignored by IR research.

The TREC conferences have heavily influenced IR research. In the last ten years it ran the novelty track, in which each sentence of a document was regarded as an information item on its own. The question answering track goes in a similar direction, here the user is not querying for a set of documents but for a concise, factual answer to his question. Both tracks show a tendency towards smaller content granularity. A fact given within a document requires on average to read half of the document until the fact is found. The *granularity* of information thus influences its retrieval costs. The coarse granularity of documents leads to high costs for locating relevant parts of documents, e. g. for re-use, aggregate queries, or question answering. The extreme case of a document is a mere list of sentences, devoid any further structure.

Data bases and ontologies allow for more structured access, i. e. browsing and searching. In contrast to documents, data bases and ontologies allow to retrieve sets of items together with relevant properties. Ontologies (and some database systems as well) allow to answer queries to which the answer has only implicitly been entered. It requires effort to structure and formalise knowledge to make it fit into a database or ontology, but retrieval abilities are also higher – ontologies are built for re-using knowledge.

Taken to its extreme, one could try to formalise all personal knowledge, leading to exorbitant externalisation costs and very low retrieval costs. In reality, one can expect a "sweet cost spot" for the total costs ($C = C_E + C_R$) as depicted in Fig. 3. For efficient knowledge work, a user should be able to work close to the sweet spot, using something half document and half ontology.

3 UNIFIED KNOWLEDGE MODEL (UKM)

In this section we present a *unified knowledge model* (UKM). It is unified in the sense that it can represent a range of existing knowledge formalisms such as documents and formal ontologies. The UKM is also unified in the sense that it represents both textual content and relations among content items. These relations can be formal or informal. The term *knowledge model* is used instead of *ontology* to emphasize a large amount of stored *textual content*.

There is a basic tradeoff between acquirability of a knowledge representation language and it expressive power (Gruber, 1989). The purpose of the unified knowledge model (UKM) is to *analyse* costs in PKM processes, not to be used by end-users for their PKM. A slightly modified version of the UKM *is* used for PKM. This version is called *Semantic Web Content Model* (SWCM) (Völkel, 2007). The main difference between SWCM and UKM is that SWCM allows different sizes of items whereas UKM has a maximal item content size of one sentence. SWCM has also more features which make it more usable for PKM.

To be able to model both documents, semantic nets and the continuum between these two models, the UKM must take into account different "degrees of formality", a notion introduced by Lethbridge (1991)[3]

Granularity is an important cost driver (c. f. Sec. 2.2). The smallest units of content that make

[3]Lethbridge uses a kind of semantic net with concept and relation subsumption hierarchies.

sense to a human are single or multiple words (encoding concepts) or sentences (encoding facts or questions). The UKM must also be able to represent formal statements. Given these two constraints, we define:

Definition. A *knowledge item* is the smallest unit of content in the UKM. A knowledge item is either

- a snippet of content which can contain something between a single word up to a sentence, or
- a knowledge item is a statement between other knowledge items. A formal statement is modelled similar to a semantic net or the RDF standard (Klyne and Carroll, 2004) as a triple of three items, the middle item playing the role of the relationship type. Representing statements as knowledge items themselves allows full meta-modelling.

A knowledge item is a technical counterpart of a knowledge cue. Knowledge cues can also be persisted in other forms, e. g. in real-world objects such as a knot in the handkerchief.

For cost analysis we need to formalise the UKM. A knowledge model K is defined as a set of nodes N and arcs A, i. e. $K := \{N, A, C, f, S\}$, where the arcs are defined as $A := \{N \times N \times N\}$. We use items as relations to allow the user to extend the vocabulary and to represent arbitrary knowledge. A function f can assign each node a piece of content from the set C of all content: $f : N \mapsto C$. A content snippet c is a simple linear stream of symbols (S) devoid further machine processable structure. We ignore aspects of natural language processing as we care about measuring explicitly structured knowledge. All structural aspects of a document or other representation format are expressed as relations between content items $c \in C$. Modelling the symbols explicitly allows to formalise the ability of the computer to map search queries via bag-of-word, vector space, or other IR-models to the content.

In the next two subsections we analyse how UKM can represent existing knowledge representation formats.

3.1 Documents

Most importantly, we analyse documents, which have been used for several thousand years now. How to model the structure and content of documents?

A French team of over 50 researchers analysed the term document in depth (Pédauque, 2003) and gives three co-existing definitions of the term "document": (i) *Document as form*, where a document is seen mostly as a container, which assembles and structures the content to make it easier for the reader to understand it. (ii) *Document as sign*, which emphasizes the argumentative structure of the content. Also, a document that can be referenced acts as a sign for its meaning. (iii) *Document as medium*, concentrates on the "reading contract", that is the intention or assumption of the author what will happen with the document.

A document contains a number of knowledge items (c. f. Sec. 3). This act of “packaging” together a set of knowledge items influences the interpretation of each item by the reader. A document is a knowledge artefact consisting of several layers. Aspects of information in a document are:

Reference-ability. Once a document is published, the reference can act as a placeholder for the content expressed within. A reference to a document can act as a meta-symbol on top of the knowledge items the document contains. The usage of document references as symbols allows a document to “participate” in conversations, which lead to scholastic methods and modern academia. To represent a document in UKM, we use one knowledge item to represent the root of the document and store the title as the content of it.

Process Metadata. Each document is written by a number of authors for a certain audience with a certain goal. By sending this process metadata along with the document the reader has the ability to put the document in context and interpret it better. Such metadata is used by the reader as a frame of reference for interpretation and for search. In UKM it is modelled as additional items linked to the document root, similar to the way how RDF is used.

Linearity. A document can typically be read from start to end by navigating through all contained knowledge items. This is modelled via an *hasNext* relation between the items holding the document sentences.

Visual Structure. A document is not only a stream of sentences, but uses type-setting, i.e. bold, italics, different font styles and size, and placement of figures. For sake of simplicity, we ignore these properties in the UKM.

Logical Structure. The visual structure is used to encode a logical structure consisting of i.e. paragraphs, headlines, footnotes, citations, and title. The logical structure makes it possible to reference smaller, meaningful parts within a document, i.e. "Sec. 4.2". Following the approach of Groza et al. (2007), we model a document structurally as a tree consisting of root, sections nested into each

other, paragraph and sentence. Sections can also contains figures and tables, which are not further modularised. In our cost model we introduce a relation *hasPart* which is used to model the different kinds of containment. To distinguish the different types of structural unit we use a relation *hasType* and a number of type-items, e. g. *section*, *paragraph*, etc.

Argumentative Structure. On top of the linear content, a document follows an argumentative structure to convey its content to the reader. Argumentative structures appear on all scales. A typical structure is the "Introduction - Related work - Contribution - Conclusion"-pattern of scientific articles. On smaller scales, patterns like "claim-proof" and "question-answer" are used. Groza et al. (2007) also describes ways to encode argumentative structures.

Content Semantics. Documents content's mean something. Building upon logical and argumentative structure, the author encodes statements about a domain within the content. We allow to store semantic statements in the UKM.

3.2 Ontologies

How to encode ontologies in UKM? A mapping from RDF to UKM is pretty straightforward. Each triple in RDF consists of URIs (U), blank nodes (B) and literals (L) and is of the form $(U,B)\times(U)\times(U,B,L)$. First we replace all literals with nodes and assign the literals content as node content, $f(n)$ =literal. Next we replace all URIs and blank nodes with nodes, using the same nodes where the same URI or same blank nodes is denoted. Now each triple is converted to the form $N\times N\times N$ and can be stored in UKM. Subtleties such as language tags and data-types of literals can be stored as further statements in UKM, so there is no information loss.

We expect future PKM systems to allow modelling textual and semantic content in the same environment, as described in (Bettoni et al., 1998; Ludwig, 2005; Oren et al., 2006)

4 A COST-MODEL FOR PKM

In this section we use the UKM to measure the externalisation and retrieval costs in PKM system. We have the following basic factors for costs and benefits:

- Each knowledge cue x that is externalised has externalisation costs $C_E(x)$. We detail these costs in the next section.
- For each task $t \in T$, the user has the option to search for knowledge cues. This has retrieval costs $C_R(t)$. Note that a user might retrieve an item several times or not at all.
- Retrieved items have a benefit $B(t)$ for the given task t.

The overall process has thus the following gain:

$$\begin{aligned} G = &\quad \textstyle\sum_t B(t) - (\sum_x C_E(x) + \sum_t C_R(t)) \\ = &\quad \textstyle\sum_t (B(t) - C_R(t)) - \sum_x C_E(x) \end{aligned}$$

4.1 Externalisation Costs

C_E can be divided into cost of authoring the content (C_A) and costs of (re-)structuring existing knowledge, classifying new or existing items or linking between items (C_S). Linking items can also be an act of formalisation if the relation is specified with a relation that has a formal semantics. Hence $C_E = C_A + C_S$.

Let N be the set of all knowledge cues in the system. Cost of content externalisation is correlated to the size of externalised artefacts. E. g. writing more words takes more time. Let $\mid n_j \mid$ be the size of the jth item, measured in the number of symbols it contains (c. f. UKM). Articulating a single symbol costs c_s. Articulating the jth item costs $\mid n_j \mid c_s$. Then $C_A = \sum_j |n_j| c_s$.

The structuring costs C_S will often involve more then one item, e. g. when linking two items. Structuring is the process of linking, tagging, typing and categorising items. The cost of restructuring are independent of the items size. The more items a knowledge base contains, the more effort it might take to find the right element to link another item to. Structuring is expected to make more of the knowledge accessible to the computer, which should enable to answer queries with better (more) results. Note: The structure of a knowledge models contains itself knowledge. It is not possible to specify the structuring costs per se, but we can expect the degree of formality of a knowledge model to correlate with the structuring costs spend. The degree of formality d_f can be measured by the amount of formal statements ($|A|$)compared to the number of knowledge items, similar to the definitions given in (Lethbridge, 1998) as $d_f = \frac{|A|}{|N|}$. If we further assume a fixed cost c_f for articulating a formal statement, we can estimate $C_S = |A| c_f$. We get the total externalisation costs

$$C_E = C_A + C_S = \sum_j |n_j| c_s + |A| c_f \qquad (1)$$

This equation assumes that no content and no formal statement is ever deleted or changed. But in reality, the cues need to be maintained to keep or improve their value over time. E. g. some knowledge is

no longer applicable or needs to be updated to reflect changes in the world. Informal ideas undergo several transitions until some of them might become text books (Maier and Schmidt, 2007).

Some structuring operations could as a side effect increase (split) or decrease (merge) the number of cues. We ignore changes to the number of cues for sake of simplicity. Deleting outdated or erroneous knowledge could improve the value of using the knowledge model quite a lot, but some costs do occur for these maintenance tasks, too. Therefore instead of measuring the knowledge model as such, we measure the costs of the operations that lead to the current state, i. e. all operations performed.

Let c be a function that assigns each operation some costs. Basic operations on a model are:

add content **(**$content_a$**)** Adding m symbols to a knowledge cue costs $m \times c(c_{sa})$ with c_{sa} being the costs of adding one symbol.

delete content **(**$content_d$**)** Deleting m symbols from a knowledge cue costs $m \times c_{ds}$ with c_{ds} being the cost of deleting a symbol. Deleting has often lower cost than adding, e. g. when deleting a complete item in the user interface which causes deletion of many symbols.

Cost of updating can be modelled as the sum of deletion costs and addition costs.

add formal statement **(**$stmt_a$**)** Adding a formal statement. The cognitive costs (measured in time and ultimately money) should vary according to the severity of the formal statement. E. g. it should take less time to create a *hasPart* or *hasInstance* relation than a *hasSubclass* statement. These differences will be taken into account in future versions of the cost model.

delete formal statement **(**$stmt_d$**)** Deleting a single statement could have dramatic effects, depending on the used inference rules.

We take the restructuring operations into account and define $n(content_a)$ as the total number of added symbols – and respectively $n(content_d)$ as the number of deleted symbols. Let τ be the total costs of an operation, calculated by multiplying the number the operation is performed with the costs of the operation, i. e. $\tau_{op} = n_{op} c_{op}$. We can reformulate the externalisation costs as

$$C_E = \tau_{content_a} + \tau_{content_d} + \tau_{stmt_a} + \tau_{stmt_d}$$

4.2 Retrieval Costs

In order to precise the relation of structures in the knowledge base and search costs one first needs to develop a unified model for the search process, which does not exist yet. A first work in this direction is the "information foraging process" (Pirolli and Card, 1995). There are three basic ways to retrieve information when interacting with an information system (Bates, 2002):

Browsing a collection of items related to the information need. In principle, two kinds of collections are possible: explicit, i. e. created by a user, and implicit, i. e. the members in the set are determined by a (semantic) query. Toms (2000) and Teevan et al. (2004) emphasise the importance of finding information "by accident", e. g. when searching for something else. Such (re-)findings are important for creative processes and knowledge creation.

Formally, browsing is the act of scanning a list of items and evaluating each of them for relevance. Evaluating a single item has the costs e. The user is free to stop evaluating items from the list at any time.

Searching denotes the process of executing a query (e. g. keywords) and refining it until the top results are relevant to the information need. Semantic queries, utilising knowledge indirectly for inferencing, also fit into this category.

Formulating a query has the costs q. Systems that allow several kinds of queries need different values for q to model the difference in cost. After each search-step the user is confronted with a list of search results which need to be evaluated, similar to browsing.

The search costs depend on the complexity of the query and the structure of the knowledge base. A complex query has a higher cost to be formulated, but has the ability to return exactly the required cue. Simpler queries return usually too many results and need refinement. From a users perspective, starting with simpler queries that are gradually refined is more economic than asking directly a complex query. The interactive refinement process gives earlier feedback about how many results are returned, which guides query refinement until the query is complex enough to filter out the desired cues. This way, queries do not become more complex than needed.

Following links should not be confused with browsing. A common practice in large search spaces for which neither suitable collections nor query terms are known is to explore e. g. citation links. Following links is thus a kind of associative retrieval.

Following a link has the costs l.

A complete search process involves typically all three kinds of operations. Instead of measuring each step, we model the complete retrieval process as a process that involves some costs $C_R(t)$. These costs can be broken down to $C_{ql}(t)$ for formulating queries and following links and costs for evaluating items.

Assuming the user evaluates $k(t)$ items in the retrieval process, we can define task-specific precision p_t and recall r_t (Van Rijsbergen, 1979). As a refinement of this idea, an relevant item has a certain benefit for the given task (in range 0 ... 1). We define the benefit of an item j for a given task t to be $v_j(t)$. For irrelevant items, $v_i(t)$ is zero.

Let $k(t)$ be the total number of all retrieved items in the search process. There is a certain cost e to evaluate each item in order to be able to decide if the knowledge represented in the item is relevant for the task. We assume the effort of evaluating a single item is not correlated to precision and recall of the complete process. The complete costs of the retrieval process for task t are then $C_R(t) = C_{ql}(t) + k(t)e$.

Only retrieved knowledge cues can bring benefit for the user. Knowledge that is never used is of zero value. The complete benefit $B(t)$ of the $k(t)$ retrieved items is $B(t) = \sum_j v_j(t)$. Both p_t or $k(t)$ might also be zero. Assuming all retrieved items in a task are either relevant (value = 1) or not, we can simplify the formula as $B(t) = p_t k(t)$.

For each retrieval process we get:

$$\begin{aligned} & B(t) - C_R(t) \\ = \;& p_t k(t) - C_{ql}(t) + k(t)e \\ = \;& k(t)(p_t - e) - C_{ql}(t) \end{aligned}$$

Thus the query formulation costs are a kind of fixed costs, whereas the relation between precision and evaluation costs decides if the whole retrieval process was worth the hassle. Interestingly, higher recall values seem in the light of cost-benefit analysis less relevant than high precision values. Precision in retrieval values is typically heavily dependent on the degree of structuredness and formality of the data.

If we analyse the equation, we see three factors than can negatively influence the gain of using a PKM system:

1. If the user does not try to retrieve knowledge for a task t;
2. if no or too few relevant items are retrieved, i. e. if the precision too low;
3. if the value of the successfully retrieved items is too low, i. e. results fit, but of too low value.

Factor (1) is addressed by (Cutrell et al., 2006) which proposes to automatically start a search when certain triggers are encountered. Factor (2) is addressed by works in knowledge articulation and modelling, information retrieval and improved search algorithms. Factor (3) can maybe only be addressed by personal experience or training.

4.3 The Complete Cost Function

Stitching the parts together we get:

$$\begin{aligned} G = \;& \sum_t (B(t) - C_R(t)) - \sum_x C_E(x) \\ = \;& \sum_t k(t)(p_t - e) - C_{ql}(t) \\ - \;& \tau_{content_a} - \tau_{content_d} - \tau_{stmt_a} - \tau_{stmt_d} \end{aligned}$$

As we see one of the most important cost drivers is the question how well the costs spend on externalisation and query formulation can improve the precision of the retrieved items. Note that most works address improvements of precision only by looking at the data as "given". In PKM, this is not true, as the knowledge items are authored and retrieved by the same audience.

5 APPLYING THE COST MODEL TO SEMANTIC WIKIS

As an example, we apply the cost model (i. e. Sec. 4.1) to a semantic wiki, in this case to *Semantic MediaWiki* (SMW) (Krötzsch et al., 2006). First we need to represent the data model of SMW in terms of the UKM. Each wiki page in SMW can be regarded as a knowledge item. SMW has two types of formal statements: Type (a) links a page to another page; type (b) links a page to a value stored within that page. We model type (a) as a formal statement in the UKM. Type (b) links can be represented as a link from the page to a knowledge item which contains that data value. For structuring, SMW allows to create semantic links or put wiki pages into categories. Categories can be modelled as knowledge items containing only their name and being linked to each category member. Addition and deletion of category links and semantic links can be measured as $stmt_a$ and $stmt_d$.

Note: As SMW uses MediaWikis versioning history, one could in theory calculate all modelling operations that ever happened. The time to externalise knowledge as text or wiki links could be measured. In the future, we consider to perform this kind of evaluation on existing public instances of SMW. Via additional usage logs one could determine the average time e. g. it takes to pose a query.

Although there has been until today no study on personal wiki use, many of our colleagues do use semantic wikis and in particular SMW as their PKM tool. SMW allows all three basic kinds of retrieval:

browsing, searching and following links. A user can browse e. g. all pages in a category, or all members of a list generated by an a semantic query embedded in a page. For search the user can either perform standard keyword search or pose semantic queries to the system. The ability to follow links is obvious in a wiki.

These properties make SMW an ideal study object for PKM, as it uses almost all imaginable ways to state and retrieve knowledge – only statements about statements are not possible in SMW.

6 RELATED WORK

There has not been much work on estimating cost and benefits in PKM.

A related work done by Bontas et al. (2006) in the area of ontology engineering does not fit our use case as the use of a personal knowledge model is not a linear, planned process with the goal of creating a formal representation. Rather the contrary is true: An individual is always reluctant to formalise anything, because its unclear if the extra effort will ever pay off.

(Lethbridge, 1998) shows metrics for concept-oriented knowledge bases, but does not take costs into account.

7 CONCLUSIONS AND OUTLOOK

This paper makes a first attempt to understand the complete PKM process in order to help design better PKM tools. The overall benefit of using a PKM system could be characterised by summarizing over the successfully retrieved knowledge items (content or formal statements) for each task. Costs could be characterised as the sum of the costs of all authoring and structuring efforts. A quantification of the effect more structuring has on lower retrieval costs (or improved benefit) cannot be stated unless the semantics of the formal statements and details of the search process (browse, search, follow links) are specified. Thus the resulting formulas can serve only as a conceptual framework or starting point for tool-specific measurements.

7.1 Future Work

As future work we intend to develop automatic measures of the information content of a knowledge model, by counting the size and used symbols (here: words) of each knowledge item as well as the number and kind of semantic links. We have to take the semantics of the knowledge model into account, as some formal statements have a much higher influence than others, c. f. ontology evaluation (Vrandecic, 2006). SMW offers only a transitive category hierarchy, hence the transitive closure can be calculated and taken into account. Counting the number and kind of modelling steps used in the history of a semantic wiki is also planned. To estimate value v_i, precision p_i, search costs c_i and number of returned items k_i in search processes, we intend to perform a diary study.

Another important aspect of future research is an investigation in which way investments in structuring (C_S) can lower the cost of retrieval (C_R), e. g. by improving p_i or k_i.

REFERENCES

Avery, S., Brooks, R., Brown, J., Dorsey, P., and O'Conner, M. (2001). Personal knowledge management: Framework for integration and partnerships. In *Proc. of ASCUE Conf.*

Bates, M. (2002). Speculations on browsing, directed searching, and linking in relation to the bradford distribution. In *Emerging frameworks and methods: Proceedings of the Fourth International Conference on Conceptions of Library and Information Science (CoLIS 4)*, pages 137–150, Greenwood Village, CO. Libraries Unlimited.

Bernstein, M. S., Kleek, M. V., mc schraefel, and Karger, D. R. (2007). Management of personal information scraps. In Rosson, M. B. and Gilmore, D. J., editors, *CHI Extended Abstracts*, pages 2285–2290. ACM.

Bettoni, M. C., Ottiger, R., Todesco, R., and Zwimpfer, K. (1998). Knowport: A personal knowledge portfolio tool. In Reimer, U., editor, *PAKM*, volume 13 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Blackler, F. (1995). Knowledge, knowledge work and organizations: An overview and interpretation. *Organization Studies*, 16(6):1021–1046.

Boettger, M. (2005). Pkm and "cues to knowledge". Technical report, 25Uhr.de.

Bontas, E. P., Tempich, C., and Sure, Y. (2006). Ontocom: A cost estimation model for ontology engineering. In Cruz, I. et al., editors, *Proceedings of the 5th International Semantic Web Conference (ISWC 2006)*, volume 4273 of *Lecture Notes in Computer Science (LNCS)*, pages 625–639. Springer-Verlag Berlin Heidelberg.

Cleverdon, C. W., Mills, L., and Keen, M. (1966). Factors determining the performance of indexing systems. Technical report, ASLIB Cranfield Research Project, Cranfield.

Cutrell, E., Dumais, S. T., and Teevan, J. (2006). Searching to eliminate personal information management. *Commun. ACM*, 49(1):58–64.

Davenport, T. H. (2005). *Thinking for a Living: How to Get Better Performances And Results from Knowledge Workers*. Harvard Business School Press.

Despres, C. and Chauvel, D. (2000). *Knowledge Horizons: the present and promise of Knowledge Management*. Butterworth-Heinemann.

Drucker, P. F. (1985). *Management: Tasks, responsibilities, practices (Harper & Row management library)*. Harper & Row.

Esser, K. B. (1998). *Ein Modell zur Verknüpfung des persönlichen Gedächtnisses mit externen Informationsspeichern*. PhD thesis, Freie Universität Berlin.

Feldman, S., Duhl, J., Marobella, J. R., and Crawford, A. (2005). The hidden costs of information work. Technical report, IDC.

Frand, J. and Hixon, C. (1999). Personal knowledge management : Who, what, why, when, where, how? Speech. working paper.

Glushko, R. (2006). 3. information organization and,or,vs search. Lecture Note.

Groza, T., Handschuh, S., Möller, K., and Decker, S. (2007). Salt - semantically annotated latex for scientific publications. In Franconi, E., Kifer, M., and May, W., editors, *ESWC*, volume 4519 of *Lecture Notes in Computer Science*, pages 518–532. Springer.

Gruber, T. R. (1989). *The acquisition of strategic knowledge*. Academic Press Professional, Inc., San Diego, CA, USA.

Higgison, S. (2005). Your say: Personal knowledge management. *Insight Knowledge*, 7(7).

Iske, P. and Boekhoff, T. (2002). The value of knowledge doesn't exist. In Karagiannis, D. and Reimer, U., editors, *PAKM*, volume 2569 of *Lecture Notes in Computer Science*, pages 632–638. Springer.

Jones, W. and Bruce, H. (2005). A report on the nsf-sponsored workshop on personal information management. report.

Jones, W., Bruce, H., and Dumais, S. (2001). Keeping found things found on the web. In *CIKM '01: Proceedings of the tenth international conference on Information and knowledge management*, pages 119–126, New York, NY, USA. ACM Press.

Klyne, G. and Carroll, J. J. (2004). Resource description framework (RDF): Concepts and abstract syntax. `http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/`.

Krötzsch, M., Vrandecic, D., and Völkel, M. (2006). Semantic mediawiki. In Cruz, I., Decker, S., Allemang, D., Preist, C., Schwabe, D., Mika, P., Uschold, M., and Aroyo, L., editors, *Proceedings of the 5th International Semantic Web Conference (ISWC06)*, volume 4273 of *Lecture Notes in Computer Science*, pages 935–942, Athens, GA, USA. Springer.

Lethbridge, T. (1998). Metrics for concept-oriented knowledge bases. *International Journal of Software Engineering and Knowledge Engineering*, 8(2):161–188.

Lethbridge, T. C. (1991). A model for informality in knowledge representation and acquisition (an extended abstract). presented at Workshop on Informal Computing, May 29-31 1991, Santa Cruz CA.

Likert, R. (1932). *A technique for the measurement of attitudes*. s.n., New York.

Ludwig, L. (2005). Semantic personal knowledge management. Technical Report D11.01_v0.01, DERI Galway.

Maier, R. (2004). *Knowledge Management Systems: Information and Communication Technologies for Knowledge Management*. Springer.

Maier, R. and Schmidt, A. (2007). Characterizing knowledge maturing: A conceptual process model for integrating e-learning and knowledge management. In Gronau, N., editor, *4th Conference Professional Knowledge Management - Experiences and Visions (WM '07), Potsdam*, volume 1, pages 325–334, Berlin. GITO.

Maurer, H. (1999). The heart of the problem: Knowledge management and knowledge transfer. In *Proc. ENABLE'99*, pages 8–17. Espoo-Vantaa Institute of Technology.

Mitchell, A. (2005). The rise of personal km. *Inside Knowledge*, 9(1).

Nonaka, I. and Takeuchi, H. (1995). *The Knowledge-Creating Company : How Japanese Companies Create the Dynamics of Innovation*. Oxford University Press.

North, K. (2007). Produktive wissensarbeit. In *5. Karlsruher Symposium für Wissensmanagement in Theorie und Prxais*. CD-ROM.

Oren, E., Völkel, M., Breslin, J. G., and Decker, S. (2006). Semantic wikis for personal knowledge management. In *Database and Expert Systems Applications*, volume 4080/2006, pages 509–518. Springer Berlin / Heidelberg.

Pédauque, R. T. (2003). Document: Form, sign and medium, as reformulated for electronic documents.

Pirolli, P. and Card, S. K. (1995). Information foraging in information access environments. In *CHI*, pages 51–58.

Polanyi, M. (1958). *Personal Knowledge: Towards a Post-Critical Philosophy*. Routledge & Kegan Paul Ltd, London.

Teevan, J., Alvarado, C., Ackerman, M. S., and Karger, D. R. (2004). The perfect search engine is not enough: a study of orienteering behavior in directed search. In *CHI '04: Proc. of the SIGCHI conf. on Human factors in computing systems*, pages 415–422. ACM Press.

Toms, E. G. (2000). Serendipitous information retrieval. In *DELOS Workshop: Information Seeking, Searching and Querying in Digital Libraries*.

Van Rijsbergen, C. J. (1979). *Information Retrieval, 2nd edition*. Dept. of Computer Science, University of Glasgow.

Varian, H. R. (1999). The economics of search. In *SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, page 1, New York, NY, USA. ACM.

Völkel, M. (2007). A semantic web content model and repository. In *Proceedings of the 3rd International Conference on Semantic Technologies*.

Vrandecic, D. (2006). Ontology evaluation for the web - phd proposal. In Diederich, J., Motta, E., and Bontas, E. P., editors, *Proceedings of the KnowledgeWeb PhD Symposium KWEPSY 2006*, Budva, Montenegro.