Research Proposal: An Adaptive, Evolutionary User Profile for Knowledge Management.

By Nikolaos Nanas
Research Student at the Knowledge Media Institute

Supervised by:
John Domingue
Stuart Watt
Enrico Motta

The Open University
June 2001

TABLE OF CONTENTS

1. INTRODUCTION

(Nanas, N. 2001) has made it clear that our general research goal is the investigation of the potential application of information filtering (IF) for knowledge management (KM). Although this direction sounds literally contradicting, the introductory discussion on the theoretical foundations of KM has identified that information is a critical resource for the creation of new knowledge at the individual level. The importance of information is also reflected by most individual-oriented approaches to knowledge management. Such KM systems use different artificial intelligence techniques to assess the relevance of information entities to the individual and/or her/his context. Relevant enough information can prove useful to the individual given her/his task at hand and can lead to "informed" decision making, which eventually leads to the creation of new knowledge.

Despite however the importance assigned to information for knowledge creation, information filtering technology has only recently emerged as an alternative to more traditional approaches to KM, like knowledge-based systems. One example of an IF-based KM system is the *Knowledge Sharing Environment* (KSE) (Davies, N. J., et al. 1998). The KSE system not only proves the applicability of IF to KM, but will also be used in the rest of this document as a general prototype of the KM services that IF can support. Section 2 structures these individual-oriented KM services in terms of the *converting* and *connecting* processes described by (O'Leary, D. E. 1998).

In (Nanas, N. 2001) we have also argued that more effective services than those in KSE can be provided if appropriate IF technology is used. Our investigation of the domain of IF has pointed towards the direction of IF systems that have the ability to both adapt and evolve. We believe that this kind of IF systems is not only appropriate for supporting KM services but can also overcome some of the disadvantages of the traditional knowledge-based approach. To support the KM services described in the next section, we present in Section 3 an architecture for the development of such an IF system, designed to target the domain of KM. The actual development of the proposed architecture and of the corresponding KM system will take place in five stages. These stages and the tests and experiments that can arise from them, are described in Section 4.

2. INDIVIDUAL-ORIENTED KM SERVICES

Our approach to KM targets the individual knowledge worker working on a knowledge intensive task as this is defined by (Borghoff, U. M. and Pareschi, R. 1998). We propose the development of a KM system that is centred on the individual and not the group, without however ignoring the importance of communication and collaboration. The system is based on an IF core and provides services that realise some of the converting and connecting processes described by (O'Leary, D. E. 1998). More specifically a user profile is used to express the information needs of each one of the individuals in a group working environment. A central repository is used to store the documents that the individuals wish to share (figure 1.). Based on this simple framework, the following connecting and converting processes and their corresponding services are supported:

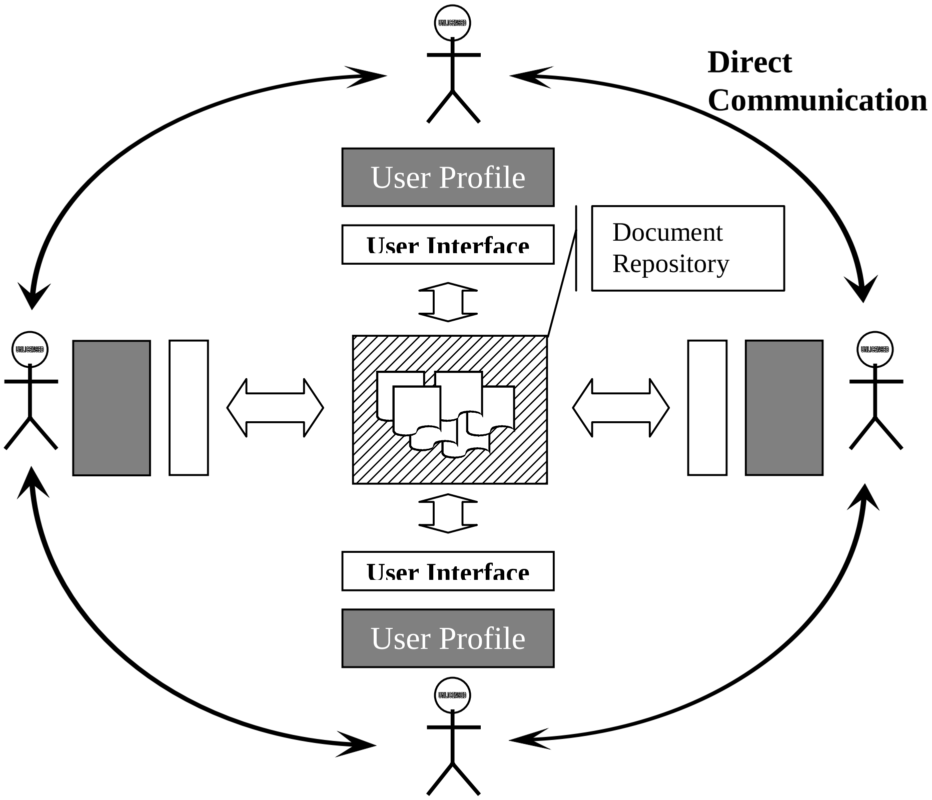

Figure 1. General system framework.

Connecting knowledge to people. The requirement for a KM system to actively provide the individual working on a knowledge intensive task with the information that she/he needs has been identified by (Abecker, A., et al. 2000, Domingue, J. and Motta, E. 2000, O'Leary, D. E. 1998). An IF-based system can realise this service in two different ways. In KSE a user is notified every time a new relevant document is added to the system’s central repository. Every new document that is supplied to the repository is matched against the profiles of the users and those that are interested enough are notified in an appropriate way (e.g. email). Another way to “push” relevant information to the user is to implicitly identify the user context. This can be done based on the document that the user is reading or working on in a particular moment (Rhodes, B. J. and Starner, T. 1996). When for example the user is reading an email or a web page the system could search for relative information and in case relative enough information exists, the user is again notified. This of course implies that the system should also have information retrieval capabilities. A combination of IF with information retrieval can be found in the *Amalthaea* IF system (Moukas, A. and Maes, P. 1998, Moukas, A., et al. 1999). The use of an explicit definition of the user context, like a process model supported by a workflow engine (Abecker, A., et al. 1999, Abecker, A., et al. 2000), is avoided for the reasons explained in (Nanas, N. 2001).

Connecting people to knowledge. The system's IR component can also support the direct retrieval of relevant information in a traditional "pool" fashion. The user can initialise a search by either forming a conventional query or by providing the system with a document. In both cases the returned broad results are filtered by the user profile before they are presented to the user. In the second case the user can search for information that is relevant to a document that he has specified. This kind of retrieval is called *document-based* retrieval and its importance in IR has been identified by (Belew, R. K. 1989). The user can thus use a web page or any other kind of document that she/he came across and found interesting, to search for existing relevant information in the repository. This service is not supported by the KSE system.

Connecting people to people. Another interesting service supported by the KSE system is the ability to search for individuals with similar interests. In this case the user's profile is matched against the profiles of all other individuals in the working group and the users with the best matching profiles are presented to the user. A similar KSE service is the identification of users that are interested in a given document. This time the user-specified document is matched against the profiles of the users and those interested enough are presented to the user. Although both these services can trigger collaboration it would be even more interesting for the user to be able to find the most appropriate individual to ask for more details on a specific document's topic. To support this service the user's interests must be distinguished from the user's expertise. A separate profile can be built to represent the user's expertise based on the documents that she/he has created. These documents can be for example the papers that a research writes or the project reports submitted by a designer. To find the most relevant expert to a document's topic the user specified document is matched against the *expertise* profiles of the rest of the users. The users with the best matching profiles are then presented to the user. Finally we believe that the above services should also be coupled with a facility for direct communication between users. This implies that the system should be either integrated with some email client and/or just enable message passing between users in the mode for example of instant messaging.

Connecting knowledge to knowledge. One basic disadvantage of the proposed IF-based KM system is that it can only deal with textual information. This problem can be however alleviated to some extent with the use of hypertext. KSE supports the annotation of documents with personal notes. We can extent this functionality to provide the user with the ability to create hypertext links between files of different type. The user can for example link an image file or a blueprint to a specific document or in the other way around annotate an image file using a text file that she/he has created or some other document. As a result the system can implicitly retrieve non-textual information as a side effect of the retrieval of the document it is linked to. In addition, it can take advantage of the *serendipity effect.* In the second case the user can use the presented documents as a starting point to browse through the linked files in a search for more relevant information. Linking documents to the artifacts produced during a task can also add task relevance to the documents (Reeves, B. and Shipman, F. 1992).

Converting individual to group-available knowledge. In (Nanas, N. 2001) we have argued that the documents that an individual creates during a knowledge intensive task reflect, at least to some extent, her/his reflection and decision process during the task. The individual's knowledge is therefore *externalised* as information that has inherently, increased capacity to prove useful under similar circumstances in the future. According to the proposed approach the user makes her/his knowledge publicly available by just submitting documents that she/he has created or that she/he deems interesting to the repository. She/he does not have to annotate the document using some formal language. As already mentioned, in KSE whenever a user submits a new document to the system the document is matched against the profiles of the rest of the users and the most interested users are notified. In the general case however the document can be stored in the repository for future use. The document is retrieved from the repository when it is found to be relevant enough to a user's information needs and context. These are expressed by her/his profile and its combination with the document that she/he is reading or working on at the moment (see above). New knowledge can now be created as the result of the individual's informed action. The created knowledge does not have to be the "same" as the knowledge that was externalised in the form of the received information. Different users can *internalise* different knowledge from the same piece of information.

The power of the proposed approach to KM comes from the interweaving of all of the above services and not from each one of them in isolation. When for example an individual is presented with a document that is relevant enough, but not exactly what she/he needs, then she/he can use the document to initialise a document-based search for more relevant documents. She/he can thus guide the system towards the information that she/he needs. Alternatively, she/he can browse the document's outgoing links in a search for more relevant information, or in a search for information that can enhance her/his understanding of the document. In the case of a document that "sounds" interesting, if the user cannot understand it to its full extent, she/he can use the document to find the most appropriate experts to ask for explanations. This service can be made more explicit if the document's author is known in advance. We can then present the document's author together with the document and thus the receiver can directly access the author for more details and explanations on the received document.

3. INFORMATION FILTERING CORE.

For the realisation of the above services we propose an architecture for the development of an appropriate user profile. We follow a connectionist approach that was motivated by the *Informer* and *PSUN* systems (Mc Elligott, M. and Sorensen, H. 1993, Mc Elligott, M. and Sorensen, H. 1994, O'Riordan, A. and Sorensen, H. 1995, Sorensen, H. and Mc Elligott, M. , Sorensen, H., et al. 1997). These system use a connectionist representation of the user's information needs that has the ability to adapt to changes in the user's information needs based on the user's feedback. However, as discussed in (Nanas, N. 2001) the two systems do not have the ability to evolve. A KM-oriented user profile has to be able to adjust to any kind of changes in the individual's information needs. This implies the ability of the profile to both adapt and evolve. Adaptive profiles can "reshape" their representation of the user's information needs and thus quickly reflect changes in them. However, only modest changes can be reflected by profiles that only have the ability to adapt. The ability of an IF system to adjust to radical changes in the user's information needs can be accomplished with evolution. Evolving IF systems maintain for each one of the individuals, a population of profiles that is spread around the information space. As a result and with the help of genetic operations, these systems have the ability to adjust to radical changes in the user's information needs. Any change of need is eventually reflected by one or more of the profiles in the population. This reaction however is not immediate. A number of system interactions with the user are usually required for a profile to converge to the new area of interest. *Amalthaea* is an example of a system that has the ability to evolve but not to adapt (Moukas, A. and Maes, P. 1998, Moukas, A., et al. 1999). In (Nanas, N. 2001) we have also described a number of IF systems that combine evolution with local learning (Baclace, P. E. 1991, Menczer, F. and Monge, A. E. 1999, Sheth, B. D. 1994, Tauritz, D. R., et al. 1999). These systems however assign greatest importance to their evolutionary component than to the adaptation at the local, individual level.

In contrast, according to the proposed architecture the profile's ability to adapt is integrated with its ability to evolve. A single user profile, instead of a population of profiles, is used to represent the user's information needs even if these comprise many topic categories. Appropriate rearrangement of the importance of the profile's components and of their interrelations allows the profile to quickly adapt to modest (local) changes in the user's information needs. At the same time evolution is achieved by purging from the profile components that has not been useful for some time, while new components that reflect new interests are added. In other words, the integration of adaptation with evolution gives to the profile the freedom to "crawl" in the feature space (figure 2.). Adaptation moves the profile quickly but locally towards the direction of the change and evolution purges components that are left behind and adds new competent components that can increase the speed of the profile's movement towards the new area of interest. In the following paragraphs we initially describe the proposed architecture in detail. We then present how the services described in Section 2 can be realised by the corresponding user profile.

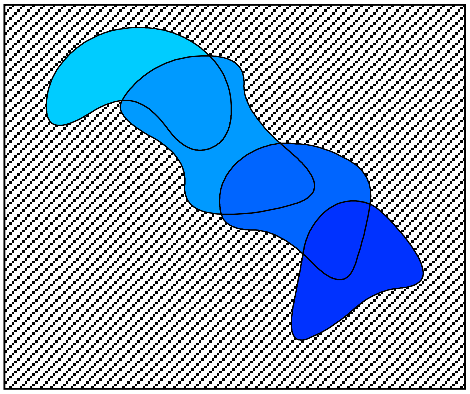

Figure 2. Profile crawling in the feature space.

Adaptation at the local level moves the profile towards the direction of the change in the user's information needs. At the same time the evolution mechanism purges the profile components that are "left behind". As a result the profile is not anchored to some area in the feature space. It has the ability to crawl in the feature space.

3.1. Profile Architecture.

A three layered representation of the user's information needs constitutes the user profile (figure 3.). The three layers are the *short-term layer*, the *long-term layer* and the *concept layer*. The layers are interrelated and collectively have the ability to represent the user's information needs and adjust to any changes in them through both adaptation and evolution. The profile is populated by concept terms extracted from documents of interest to the user. Each term is represented in the profile by a node. As it will be explained in detail later, the terms are spread among the three layers according to their consistency of importance in representing the user's information needs. Each term is assigned a weight corresponding to its statistical importance in the relevant documents. Terms are also connected with weighted links based on their context in the documents that they appear in. To present the architecture in more detail we describe in the following paragraphs the initialisation of the profile, its evolution, the way it evaluates documents, and its adaptation.

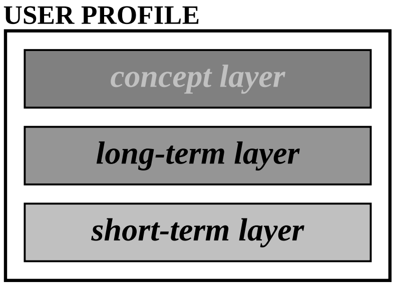

Figure 3. Three-layer representation of the user information needs

3.1.1. Profile Initialisation.

To initialise the profile the user is asked to submit a number of documents that are representative of her/his interests. This is the only effort the user has to put in generating her/his initial profile. The user-specified documents are processed for selecting the most important terms to be added in the profile. Dimensionality reduction is initially achieved by removing stop words and by stemming words with common root. Further reduction is achieved by identifying concept terms in the user specified documents. These terms can then be used to populate the profile.

To select the most important terms to be added to the profile we have experimented with a number of term weighting methods (for details see the pilot study). In (Nanas, N. 2001) we have expressed the requirement for term weighting methods that only use relevant documents in order to assess the importance of terms. We have experimented with a number of such methods. According to the results the most promising method is based on a combination of the term's document frequency in the sample of relevant documents and the term's document frequency in the complete collection of documents stored in the repository. More specifically a weight is assigned to each term in the relevant documents using the following formula.

$W_t=df_t/N-cdf_t/NC$

Where:

- W_t is the weight of term *t*.
- df_t is the term's document frequency in the sample of relevant documents.
- cdf_t is the term's document frequency in the complete collection.
- *N* is the number of relevant documents.
- *NC* is the number of documents in the repository.

The essence behind the above term weighting formula is that terms that are related to the topics of interest to the user will appear with a greatest ratio in the documents of interest to the user than in the complete collection. The formula distinguishes as good discriminators those terms that appear in most of the relevant documents but only in some of the documents in the complete collection. The discriminating power of *document frequency* was also identified by (Greiff, W. R. 1998, Yang, Y. and Pedersen, J. O. 1997).

In addition to its conceptual simplicity, the formula has a number of advantages. First of all it is based only on relevant documents. It is not dependent on the user providing negative examples. Furthermore since both df_t/N and cdf_t/NC have a value in the interval [0,1] the calculated term weight ranges between –1 and 1. Of course terms with negative weights are excluded from being added to the profile. As a result the weights of the terms that are added to the profile have the important characteristic that their value ranges between 0 and 1. Another advantage of the formula is that updating the value of the cdf_t and *NC* variables is a very straightforward and computationally cheap process. A list of all the unique terms in the collection and their corresponding cdf_t can be maintained. Every time a new document is added to the collection the values in the list can be updated by just increasing the cdf_t of the terms in the list that also appear in the new document by one. Terms that are not yet included in the list are added with an initial $cdf_t = 1$. Of course, *NC* is also increased by 1. This *incremental* mode of updating the variables involved in the calculation of the term weights is advantageous in comparison to the *batch* mode of other methods like *tfidf*. The calculation of *tfidf* term weights for example would involve analysing the whole collection each time a new document was added to it. Finally the calculated weights are document independent. The calculation is based on set of documents and not on the term's statistical importance within individual documents.

Unfortunately the proposed term weighting method has a drawback. It works fine as long as the documents that the user provides for the initialisation of the profile do not span many topic categories. A problem arises when for example the user provides ten documents from ten different topic categories of interest. For more details on this disadvantage see (Nanas, N. 2001). One solution to the problem would be to ask the user to provide documents that she/he has pre-classified according to some general topics of interest. Although this is a technique adopted by many existing IF systems we would prefer to avoid implying this extra burden to the user. Alternatively, we investigate combinations of the above measure with methods that measure the importance of a term in the complete collection irrespectively of defined topic categories. One such measure is the *ResidualIDF (Church, K. W. 1995).* Nevertheless, as it will be explained bellow, the term weighting method used is not critical to the success of the system. It is used as a first filter of the importance of terms and it has just to be good enough in order for the initial profile to work satisfactorily. One problem with any adaptive system is the *cold start* phenomenon. Users have to be satisfied enough by the system's initial performance to be motivated to use it.

After the terms are weighted, the best *N* terms are selected to populate the profile. *N* depends on the stage of system usage. For initialisation *N* corresponds to the number of terms allowed to populate the profile. This parameter has either to be fine-tuned for better system performance or to be specified by the user. The selection of terms is followed by a secondary analysis of the relevant documents for the creation of links between the selected terms. In *Informer* and *PSUN* terms are linked according to their appearance in the same phrase. We however believe that a more powerful representation of the user's information needs can be generated if the selected terms are linked according to their pattern of occurrence in the relevant document. We thus treat a document as a string of words without taking into account any phrase delimiters. A link is created between two selected terms if they appear "close enough" in a document. The link is then weighted according to the "distance" between the terms. One way to achieve this kind of linking is by using a "window" of *W* terms that is moved through the document. More specifically the proposed algorithm can be described as follows:

1. Start from the first word in the document.
2. If the word is not a selected term move to the next word until you find a selected term.
3. When a selected term is found start counting the number of words that follow it.
4. If in less than *W* words another selected term is found then link the two selected terms with weight 1/*n*, where *n* is the number of words that intervene between the two selected terms. Move to the second selected term and go back to (3).
5. If there is not a selected term in the "window" of *W* terms move to the *W+1* term and go back to 2.
6. Continue until the end of the document.
7. If a link between the same two selected terms has been generated more than once use the average of the weights.

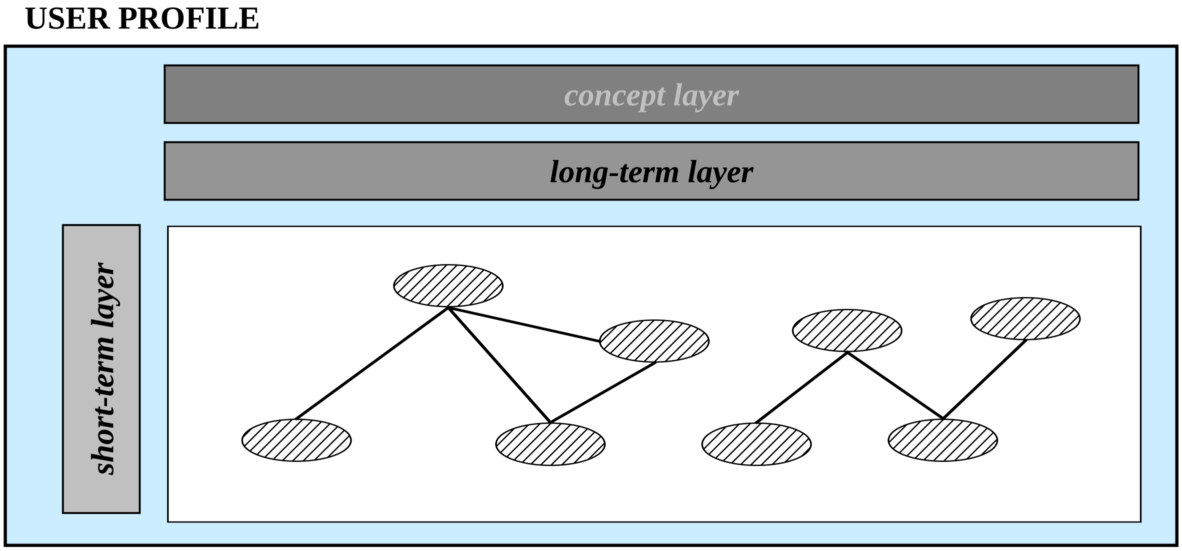

Figure 4. Initial User Profile.

The proposed algorithm generates bi-directional links with weights between 0 and 1. If two selected terms are next to each other then they are linked with a weight equal to 1. In this way the algorithm has the ability to distinguish collocations between concept terms. More importantly however the links represent the probabilistic dependency between sets of terms and the fact that a document is relevant to the user (explained later). Variations of the above algorithm will of course be investigated. The above algorithm just demonstrates a way in which links between concept terms can be generated by taking into account their patterns of appearance in each of the relevant documents as a whole and not in individual sentences.

Once the important terms have been selected and the links between them have been generated the terms and their corresponding links are added to the profile's *short-term* layer. This is the only layer that allows the addition of new terms in the profile. The initial profile has thus the form depicted in figure 4. Although the *long-term layer* and *concept layer* are empty at this stage of the profile, the latest is nevertheless functional. The user can start using it in the ways described above.

3.1.2. Profile Evolution.

To achieve the evolution of the profile we propose the use of an economic model, similar to the ones used by (Baclace, P. E. 1991, Baclace, P. E. 1992, Menczer, F. and Monge, A. E. 1999, Moukas, A. and Maes, P. 1998, Moukas, A., et al. 1999). According to this model each one of the terms that are added to the profile receives the same quantity of *"birth money"*. Every time the profile is used to filter documents the profile's terms pay some of their money as rent for "inhabiting" the profile. The rent that each one of the terms has to pay depends on the layer it belongs to. The higher the layer the less the rent a term has to pay. Now whenever a term appears in a relevant document it is rewarded with extra money. In contrast when a term appears in a non-relevant document then it is penalised to pay extra money. In this second case the system takes advantage of negative feedback without however being dependent on it. This is because the money of non-competent terms is reduced due to their obligation to pay rent every time the profile is used. Terms that do not appear in documents that the user has deemed interesting or useful (positive feedback), are not paid extra money and as a result their money is eventually reduced. The *P* "poorest" terms can now be purged from the profile. *P* depends on the number of available positions in the profile and as it will be explained later, on the change in the system's performance.

Obviously the effect of the above economic model is twofold. It identifies non-competent terms that either no longer represent concepts of interest to the user or that were a bad choice in the first place. In the first case the evolutionary mechanism gives to the profile the flexibility to move in the feature space by removing from the profile any components that no longer participate in the identification of relevant documents. *INFOrmer* and *PSUN* lack this ability and therefore their profile representation is anchored to its initialisation area. In the second case the evolutionary mechanism acts as a secondary filter of function words that were mistakenly identified as important during document pre-processing. In other words terms that were mistakenly selected by the term weighting method will be eventually removed from the profile since they will prove incompetent in identifying documents of interest to the user.

Most importantly however the proposed economic model has the ability to identify terms that exhibit consistency in representing concepts of interest to the user. These terms are metaphorically awarded with higher “rank”. More specifically terms that appear “long enough” in the *short-term layer* are promoted to the *long-term layer.* As a result the initial profile takes progressively the form depicted in figure 5. Existing links between terms are maintained even between terms of different rank. As a result the two layers become interrelated. As it will be explained in the next section, the interrelation between layers plays an important role during document evaluation.

USER PROFILE

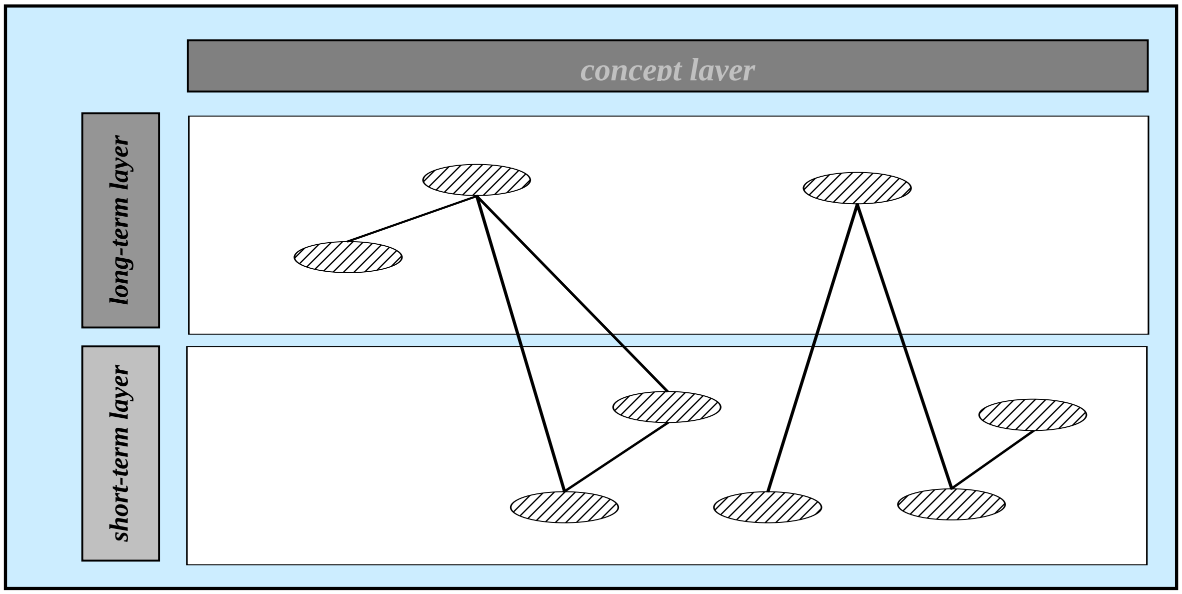

Figure 5. Consistently important terms are promoted from the *short-term layer* to the *long-term layer*.

A similar strategy is followed for promoting a term from the *long-term layer* to the *concept layer*. Candidates for promotion are once more terms that have survived “long enough” in the *long-term layer*. In addition however, a term is promoted to the *concept profile* if it is included in either the *long-term layer* or *concept layer* of the profiles of a number of other users. In essense this last mechanism adds a collaborative filtering flavor to the profile. In contrast to the WWW which is the target domain of most IF systems, group working environments are less ambiquous. There are always similar information interests and/or needs between colleagues, which are implied by their common goals, expertise, projects, etc. It is also less likely that there will be problems of polysemy. The meaning of words is usually disambiquated by the context of the organisation. The meaning of the word “rock” for example is clearly defined in the context of a company designing rock climbing equipment and would never be confused with the corresponding music scene. Conclusively terms that are promoted to the *concept layer,* represent concepts defining general categories of interests in the organisation. There will of course be some overlapping between the terms in the *concept layer* of different user profiles, depending on their position in the organisation, their expertise, and the projects that they are involved with. The final user profile takes the form depicted in figure 6.

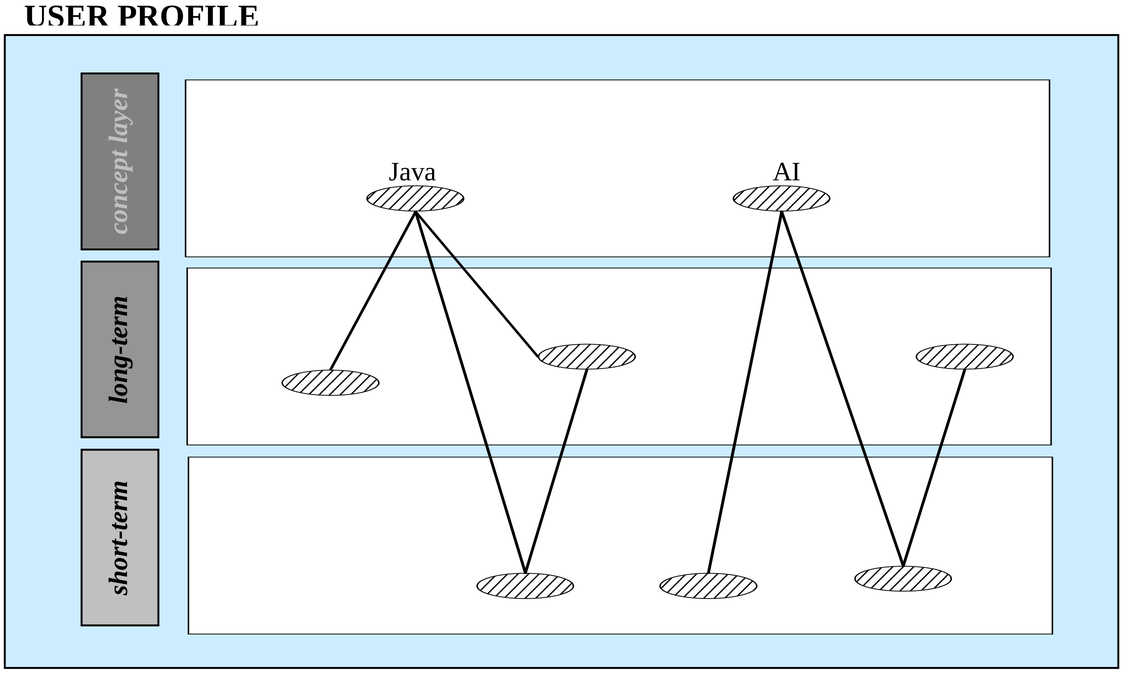

Figure 6. Consistently important terms for more than one users are promoted to the *concept profile*.

In general we expect that terms representing higher-level concepts of interest to the user will be promoted to higher profile layers. These kind of concepts correspond to higher level topics of interest to the user that remain relatively stable. A Java programmer for example will always be interested in documents about Java. What changes is the different subcategories, like for example java technologies or java applications, that depend on the programmers current context (e.g. project). This distinction between relatively stable higher level concept categories and more frequently changing lower level subcategories is reflected by the proposed user profile architecture. As already mentioned the rent that each term has to pay decreases as its rank becomes higher. Terms in the *long-term layer* pay less rent than terms in the *short-term layer* and of course terms in the *concept layer* either they never pay at all or they pay less than terms in the *long-term layer*. We can also relate the behavior of the proposed representation to the short-term and long-term human memory. The *short-term layer* acts as a term evaluation tool. Terms are purged and added more frequently than for the rest of the layers. It is also the only layer that has direct contact with the "outside world". Only terms that remain in the *short-term layer* (short-term memory) long enough are stored in the *long-term layer* (long-term memory). After a term reaches the *long-term layer* it is more difficult to be forgotten. In contrast terms in the *short-term layer* struggle for survival. Finally, terms in the *concept layer* have riched a higher level of conceptuality. They not only represent concepts of interest to the user but also concepts that define the general areas of interest within the working group.

Of course, in order for the proposed economic model to exhibit the behaviour described above we will have to appropriately specify its parameters. This fine-tuning of the parameters involved in the presented economic model is going to be performed during the testing involved in the corresponding development stage (see section, "Testing the Evolutionary Mechanism").

3.1.3. Document Evaluation.

The adopted term weighting method assigns a weight *Wt* to each one of the terms in the processed documents. This weight corresponds to the term's discrimination power, i.e. its ability to distinguish documents useful to the user. Whenever a term appears in a document under evaluation it fires and as a result it contributes its weight to the document's relevance. If for example a term appears *n* times in a document *d* then its contribution to the document's relevance will be $C_d=n*W_t/norm(N)$, where $norm(N)$ is some normalisation based on the number of terms in the document. The most interesting part however of the document evaluation process comes from the interrelation between layers. After the contributions of the terms in the *short-term layer* are calculated, part of a term's contribution is also passed to the terms in the higher level layers that it is connected with. If for example a term's (*t1*) contribution is C_{t1} and the term is connected to another term (*t2*) in a higher level layer then the contribution $C_{t1t2}=wl_{t1\text{-}t2}*C_{t1}$, where $wl_{t1\text{-}t2}$ is the weight of the link between the two terms, is passed to *t2*. The overall contribution of a term in either the *long-term layer* or the *concept layer* is thus calculated as $C=C_d+\Sigma C_t$, where C_d is the term's contribution according to its appearances in the document and ΣC_t is the aggregate of the contributions that the term receives from terms in lower level layers. Of course the overall contributions of the terms in the *long-term layer* have to be calculated before the overall contributions of the terms in the *concept profile*.

Another interesting characteristic of the way documents are evaluated is the inherent ability of the proposed representation to categorise documents. If for example the user is interested in two distinct topic categories then it is more likely that concept terms that appear in the documents of the one category will not appear in the documents of the other. Therefore, there will be either not any or very few links between terms corresponding to these different topic categories. As depicted in figure 6, two distinct graphs have been formed, each one representing one of the topic categories, e.g. "java" and "AI". If now a document about java is evaluated by the profile then naturally, terms in the "java" graph have a greater probability to fire than terms in the "AI" graph. As a result the higher level concept "java" will receive more contributions from its "subordinates" in lower level layers than the "AI" concept. The overall contribution of "java" will thus be greater than the overall contribution of "AI", and the system can inform the user that the particular document is about java. Conclusively, a single user profile has the ability to learn the categories of interest to the user and inform him about the topic category of each individual document. We should also note that the identification of a document's category can be achieved even if the higher-level term representing the category, e.g. the term "java" does not appear in the document. If the "java" graph is activated enough then the contributions that the "java" term will receive from terms in the lower level layers will be enough to identify that the document is about java despite the fact that the term "java" does not appear in the document.

3.1.4. Profile Adaptation.

So far we have described the profile’s evolutionary mechanism which results in the purging of non-competent terms and the promotion of competent ones to higher levels. As a result the profile representation evolves to finally acquire the form depicted in figure 6. At every stage of this evolution the profile is functional and can be used to filter documents. Furthermore at every stage the profile is adaptive. Adaptation is achieved by appropriately modifying the weights of the nodes and the links in the profile and is based on the user’s feedback to the presented documents. The user can either declare that she/he was satisfied by a result (positive feedback) or that it was not what she/he was looking for (negative feedback). Once more we avoid the use of scaled feedback to reduce the required user interaction with the system. Positive (negative) feedback on a document means that the profile has to be moved towards (away of) the position of the document in the feature space. How much the profile will be moved depends on the rank of the document in the presented list. If for example a document was assessed by the profile as the most relevant document and the user has given negative feedback on it, this means that the profile has to be radically moved away from the document. In contrast a document presented later in the list that is the first document in the list to receive possitive feedback, signals that the profile has to be radically moved towards the documents position in the feature space. In other words the effect that the user feedback should have on the profile depends on the certainty with which the profile has presented the document to the user.

Based on the previous rule of thumb we can construct an algorithm that assesses “how much” the profile should be moved towards or away of a certain document. We can express this quantity as a percentage of the relevance that the profile has assigned to the document. We can then disseminate backwards this amount in the profile to appropriately modify the weights of the terms and of the links that got involved in evaluating the document. To find the most appropriate algorithm we currently look in the domain of Neural Networks and investigate the potential application of algorithms like the backpropagation algorithm.

Once the documents have been presented to the user and she/he has given feedback, the relevant documents are analysed, as in the case of profile initialisation, for extracting new terms to be added to the profile. Only important terms that are not already included in the profile can be added. The number of terms to be added to the profile depends on how much we want the profile to move in the feature space. Addition of many new terms forces the profile to quickly jump to a new position in the feature space. Such a radical change of the user profile can be useful if a sudden drop of the profile’s performance has been identified. Conclusively we can calculate the number of terms to be purged from the profile according to changes in the overall system performance. The terms that are purged are then replaced by the terms extracted from the relevant document. Finally if the documents that the user has evaluated as relevant are not enough we can store them until a satisfactory amount of relevant documents has been collected.

3.2. Realisation of the proposed KM services.

In this section we describe how the KM services desrcibed in Section 2 can be realised by the proposed profile architecture. More specifically:

Connecting people to knowledge. As it was explained in Section 2, the user profile has also to be able to provide information retrieval functionality. Providing to the user the ability to form a conventional query is of course straightforward. The only requirement in this case is that the documents in the repository are indexed using for example an inverse index. Existing programs can be used to accomplish the indexing which can also be integrated with the calculation of the parameters df_t and *NC* for the complete collection (see paragraph on term weighting). We should also make sure that the results are broad enough so that it is the IF component that actually selects the documents to be presented to the user. When the user performs a search using a conventional query the broad returned results are filtered by the user profile and the documents are then present to the user with decreasing order of relevance.

In the case of *document-based* retrieval a query is formulated based on the overlapping between the user-specified document and the user profile. The query is constructed by those terms in the document that are also included in the profile. Furthermore a spreading activation function is used to temporarily activate profile terms that are also included in the document and also those terms in the profile that are directly linked to them. As a result the profile is temporarily moved towards the position of the specified document in the feature space. This of course affects the evaluation of the documents that are returned by the formulated query. The profile is now more sensitive to documents that are relevant to the specified document. In essence the user has the ability to direct the profile towards a narrower area in the feature space that corresponds to its temporary information needs. After the completion of the interaction cycle, i.e. after the user has read and evaluated some of the presented documents, the profile terms are deactivated and the profile returns to its initial state.

Connecting knowledge to people. The profile can also be used to evaluate each new document that is added to the repository. The process is straightforward but it implies the use of some kind of threshold for the profile to decide if the document is relevant enough to be presented to the user. This threshold can be learned based on the interaction of the user with the system. Whenever the user evaluates a list of presented documents, if the user has evaluated as relevant at least one of the five most relevant documents then the threshold is set to the similarity measure of the fifth most relevant document. If the user has not evaluated as relevant any of the five most relevant documents then the threshold is set to the similarity measure of the first document that the user has evaluated as relevant. The threshold can be also modified according to the user's feedback to the presented new document. If the feedback is negative then the threshold is increased and vice versa. Finally we can allow the user to have the ability to modify the threshold himself according to the number of new documents she/he wants to be receiving.

A threshold is also needed when the profile actively searches for relevant documents. If the user spends "enough time" reading or working on a document the profile's active mode is triggered. In other words the time spent on reading or editing a document can be used as implicit feedback on the relevance of the document (Kim, J., et al. 2001). The process that is followed is exactly the same as with *document-based* search with the difference that the used document is specified by the system and not by the user. In this case the use of the activation function has the effect of making the profile take under consideration the user's context as it is defined by the document she/he is reading or working on. We should also note at this point, that given the frequency with which terms are purged and added to the *short-term layer*, the latest always reflects the content of the documents that have been useful to the user in the near past. This is another way of implicitly taking into account the user's context.

Connecting people to people. To compare the profiles of two users, one of the profiles is treated as a document containing the terms in the profile. The weights of the terms in this second profile replace the terms' frequencies in the hypothetical document. The contribution of a term in the first profile can thus be calculated as $C=W_{tp1}*W_{tp2}$, where $W_{tp1(2)}$ is the weight of the term in the first (second) profile. No normalisation has to be used since the profiles have the same number of terms. Alternatively, we can neglect the links between terms and treat the profiles as weighted keyword vectors. The traditional cosine similarity measure can then be used to compare the two profiles. According to the above functionality the user can search for other individuals with similar information interests and needs. It is interesting to note at this point that this can be done by using the whole profile or by specifying certain topic categories. A user interested in "java" and "AI" for example, can search for other individuals that are interested in "java" by specifying the corresponding higher level concept in the *concept layer*. In this case the nodes in the lower level layers that are implicitly or explicitly linked to the "java" concept are activated using a spreading activation function and the comparison proceeds in the same way as described above. In essence the user can activate the area of his profile that corresponds to his interest in "java" and thus find who else in the organisation is interested in "java" and the subtopics of "java" that he is interested in. Another solution to searching for users with similar interests by topic, is to just use the terms in the *concept layer*. In this manner if the user asks for all users that have an interest in "java" then the system will return all these users with "java" appearing in their *concept layer* and with a significant weight. Although this solution is simpler it has the disadvantage that it will return users that are interested in "java" even if their subtopics of interest are different to the subtopics of interest to the user.

To identify individuals that are interested in a given document, the document is compared to the profiles of the individuals in the way we have already described. If an *expertise* profile is used in addition to the interest profile, the user can use the latest process to identify individuals that can provide him with further details on the document's content. In all of the above cases the individuals are presented to the user with decreasing order of similarity to either her/his profile or to the specified document. No threshold has to be used.

Connecting knowledge to knowledge. To support the creation of hypertext links between documents, hypertext functionality has to be added to the system. This addition does not affect the design of the profile's architecture. It however provides the user with an additional information seeking strategy. The links that have been created between a document and some other information entities can be presented together with the document itself. The user can thus use the document as a starting point to navigate in the hypertext space in a search for more useful information.

Converting individual to group-available knowledge. Every document that is stored in the repository is publicly accessible. Thus whenever a user wants to share a piece of information she/he just has to submit it to the repository. This piece of information can be either a document that the user has created as part of her/his knowledge intensive task or a document that she/he found interesting or useful. Non-textual information can also be submitted to the repository if it is linked with some textual piece of information that could enable its retrieval. The fact that the user does not have to annotate the documents that she/he wishes to share in terms of some formal representation minimises the effort she/he has to put in sharing her/his knowledge.

3.3. Advantages of the proposed IF-based approach to KM.

In general the power of the proposed approach derives from the flexibility of the profile architecture that is used to express the user information needs and from the alternative ways that the user can use to interact with the system. The proposed architecture's flexibility allows the user profile to adapt to short-term changes in the user's information needs and also to follow any longer-term and more significant changes in the feature space (evolution). The ability of the profile to efficiently adjust itself to any kind of changes has an additional effect. The user context is at least implicitly reflected by the representation. Terms in the *short-term layer* represent the content of the documents that the user has been consulting in the near past, the general topic categories of interest are represented by terms in the higher level layers and the terms in the *concept layer* represent the user's niche in the topics of interest in the organisation as a whole. At the interaction level the user can choose between a number of alternative ways for searching for relevant information. The user does not have to construct a query using some formal language. She/he can either construct a query in a traditional way, as she/he is accustomed with from her/his interaction with existing search engines, or she/he can initialise a *document-based* search. In the second case the user can use a document (or even a fragment of a document) to guide the system towards a specific information need. Thus although the initial results can be not exactly what she/he was looking for she/he can use one of the presented document to identify to the system her/his exact information needs. The fact that the results can be imperfect is also alleviated by the user ability to navigate the document space using a presented document as her/his starting point. In this way results that were not understood well enough can become more clear based on the information entities they are linked with. More explanations on a specific piece of information can also be provided by other users whose expertise is relevant to the information.

Finally, the proposed profile architecture is independent of the document structure or type. Any kind of document can be analysed as long as the system can have access to its content. This of course implies the need for an extra layer that converts incoming documents from their original format (e.g. doc, ps, pdf and html) to ASCII text.

Furthermore, if prior knowledge on the document structure exists, then it can be easily incorporated in the way the weighting of terms is performed. For example terms in the title can be treated with preference. The only design decision that is dependent on the domain of application is the user interface. The interface has to be appropriately selected to fit in the working habits of the individuals. As the following section will reveal we are going to base this user dependent design decision on feedback from real users.

4. FIVE STAGES OF DEVELOPMENT

In order to implement the proposed IF-based KM system we distinguish between five different stages of development. The first four are related to the implementation of the IF core and its testing for assuring satisfactory filtering performance. The fifth refers to the encapsulation of the IF core with an appropriate user interface which will enable the actual testing of the KM system in a real environment. The five stages of development are:

4.1. Assuring satisfactory initial performance.

As already mentioned most adaptive systems suffer from the problem of difficult user adoption at the initial level, that is known as the *cold start* phenomenon. An adaptive IF system for example, has to substantially learn the user preferences before it is able to perform satisfactorily. The system's initial performance nevertheless, has to be good enough to attract the user in using the system and not disappoint him/her with bad initial results. In the second case the user will be discouraged to use the system and as a result s/he will never provide the system with the information that it needs (feedback) in order to learn her/his information needs or interests. Special care has thus to be taken during system initialisation so that the "seed" profile performs satisfactorily enough during the early stages of the system's usage.

In the case of the proposed architecture the profile initialisation is based on user specified documents. The selected documents have to be representative of the general topic categories of interest to the user. As it is revealed by (Foltz, P. W. and Dumais, S. T. 1992), it is more efficient for the users to express their interests in terms of whole documents than in terms of isolated words. The term weighting method that we have described selects the most important terms in the selected documents based on their calculated weights. After the links between the selected terms has also been generated, both terms and links are added to the *short-term layer* of the profile. Since the higher level layers are not still occupied the links between the terms do not play any role during document evaluation. A term passes some of its contribution only to terms in higher profiles that it is linked with. Conclusively, the assessment of the relevance of retrieved documents is based only on the term weights. This makes the calculation of the initial weights critical for satisfactory initial performance.

So far in our research, we have evaluated the performance of the proposed term weighting method both empirically by observing the extracted terms for each topic category and also in terms of the filtering performance of the initial profile that was constructed based on the extracted terms (Nanas, 2001a). The results in both cases are very promising. However, as already mentioned, the proposed term weighting method performs well, if each topic category of interest to the user is represented by a substantial number of user-specified documents. To solve the problem we investigate combinations of the proposed term weighting method with task-independent weighting methods. Of course such an addition will also give rise to comparisons between the single and the combined weighting method. We also plan to study the characteristics of user-specified documents. A number of users can be asked to provide a collection of documents each that represents her/his interests. The documents can be then analysed for identifying the characteristics of the collections. Do users specify many documents for each one of their topic categories of interest or do they supply one document for each out of many non-overlapping categories?

4.2. Dependence vs. Independence.

As we have discussed in (Nanas, N. 2001) most IR models assume statistical independence between terms. Although the majority of researchers agrees that this assumption is false, the simplicity that it offers (less parameters have to be calculated) makes it quite attractive. Some approaches to IF have tried to incorporate dependency in their models (Cohen, W. W. and Singer, Y. 1996, Krulwich, B. and Burkey, C. 1997, Krulwitch, B. 1995), but the followed solutions usually just keep track of whole sentences in order to capture the dependence between terms. Connectionist approaches have the ability to represent dependence in a more flexible way. The interconnection between terms allows the representation of various term combinations (e.g. phrases) by the same terms (Mc Elligott, M. and Sorensen, H. 1993, Mc Elligott, M. and Sorensen, H. 1994, O'Riordan, A. and Sorensen, H. 1995, Sorensen, H. and Mc Elligott, M. , Sorensen, H., et al. 1997). Our approach adopts the latest solution for representing dependency. Furthermore, the proposed profile architecture has the advantage that it can function either by assuming or by not assuming dependence. Links between terms are activated, i.e. they can pass contribution, only when they span different layers. In other words independence is assumed within layers and dependence can only exist amongst layers.

Obviously in the initial profile no dependence between terms is represented (activated). After the profile has reached its final form, i.e. all layers have been populated, we can choose to incorporate dependence (allow passing of contributions) or to just use the profile without taking dependence into account (links between terms are ignored). We can thus test if representing the dependence between terms increases the system’s performance. We can directly construct a user profile with its final form by for example choosing the best A% of the terms to be directly added to the *concept layer*, the next B% (B>A) to populate the *long-term layer,* and the rest C% (C>B) for the *short-term layer*. The artificially constructed profile can be then tested in both the above modes, for comparing the effect of dependence to the system’s performance.

4.3. Testing the Evolutionary Mechanism.

To test the effect of representing the dependence between terms we can artificially construct a profile with all its layers populated by terms (see above). In the real situation however, this profile structure will be the result of the profile’s evolution, and more specifically of the promotion of constantly important terms to higher level layers. As explained in the corresponding section, the profile’s evolution is based on the user feedback. To avoid prematurely committing ourselves to real user experiments we can alternatively test the evolutionary mechanism using artificial users. We can replace real users with agents that have the responsibility of giving feedback to the system. Each agent specialises in a number of topic categories. An initial profile is constructed based on a percentage of the documents in each category. The profile is used to filter the documents and the agent evaluates the results according to its pre-specified preferences. Hopefully, the higher-level concept terms for each one of the categories will be progressively promoted to the higher level layers. Initially, the agents can be set to be interested in only one topic category. We can then add a new topic category to the agents’ preferences to simulate the emergence of a new area of user interests in the real situation. In this case the profile will have to be able to reflect the change by incorporating new terms that represent the new interest. Furthermore, the initial interest can be removed to test if the profile is able to forget topics that are not interesting any more. The evolutionary mechanism has to be able to purge from the profile the terms that represent this obsolete information interest.

The above testing methodology and evaluation criteria will be used to appropriately specify the parameters involved in the proposed economic model. Although the actual quantities of “birth money” or “rent” are not important since everything is relative, it is important to assure that the model exhibits specific characteristics. For example we should make sure that terms that are added in the profile leave “long enough” on their “birth money” so that they are given the chance to appear in a relative document and thus prove useful. In other words we should define the rent in such a way that even if a term does not appear in relative documents it will “survive” for a number of filtering sessions. Furthermore, we should appropriately specify the amounts of money with which terms are rewarded or penalised. In this case care should be taken so that frequent functional words that appear in a lot of documents and that have been mistakenly added to the profile, are penalised with such an amount that the overall effect will be the decrease of their money and their purging from the profile. Finally we should note at this point that since the involved amounts are relative we could always be normalising them to avoid a constant increase and instead keep the largest possible amount of money that a term can possess constant. Conclusively, the above testing approach will help us specify an appropriate equilibrium between the amount of money that a term possesses and the amount with which he is penalised or rewarded.

4.4. Testing the ability of the system to adapt.

So far we have described how we are going to test the initial system’s performance, the effect of representing dependence between terms in the form of links and the ability of the evolutionary mechanism to identify terms that are not competent any more and terms that are constantly important in representing the user interests. To test the effectiveness of the evolutionary mechanism we ignored any adaptation of weights. The opposite can be done to test the system’s ability to adapt. We can first directly construct a profile in its final form in the way we did to test the effect of dependence. The profile’s performance is initially tested without any feedback. The same artificial users can then be used to provide feedback to the system and thus the weights of the nodes and the edges can be appropriately modified. The performance of the adapted profile can then be measured and compared with the performance of the initial profile. Additionally we can once more start with agents that specialise to one topic category, then add one more and finally remove the initial interest to the first category. Since the evolutionary mechanism is deactivated, the profile does not have the ability to incorporate any new terms. So despite the fact that adaptation will try to stretch the profile representation towards the direction of the new area of interest, we expect that the system's performance will fall. The profile does not have the ability to follow radical changes in the user interests in the feature space.

Conclusively, the above tests will reveal that adaptation facilitates the learning of the user interests and alleviates any deficiencies of the weighting of terms and weights. Adaptation is however unable to reflect radical changes in the user interests. Evolution is required in this case. It is the combination of both adaptation and evolution that will provide the profile's ability to adjust both to quick local changes and to radical changes like the emergence of a new area of interest in the feature space. To test the combination and collaboration between all four stages of the profile's flexibility, i.e. term weighting, dependence between terms, evolution and adaptation, the complete profile can be tested using the same artificial users. We should also note that the proposed tests are going to be performed using some standardised corpus, which we are currently investigating.

4.5. Encapsulating the IF core in a KM system.

The transition from the IF core to the actual KM application involves the development of an appropriate user interface. The adopted interface has to tap into the existing work practices. If for example the users work mostly with word documents then the adopted interface can be integrated into the existing word processor. In the case of web based collaboration the interface would of course be web based. Another solution that we investigate is to adopt the interface design used by instant messaging (IM) applications. The direct communication between individuals that is provided by IM applications can be enhanced with the proposed KM services.

A number of overlapping user studies is going to be used to acquire the requirements of the KM front-end. Furthermore the studies are going to be used to provide evidence in support of certain design decisions and arguments that lead to the proposed approach. In most of the cases a bottom-up approach is going to be used. Structured questionnaires will be avoided. Instead we are investigating simple ways of collecting a substantial amount of raw data by following a kind of "ethnographic" approach. All of the studies are going to be conducted within the academic environment of the Knowledge Media Institute (www.kmi.open.ac.uk). More specifically the investigated methods and the corresponding goal or goals are:

Mapping of tasks to document usage. The goal of this study will be to identify basic higher level knowledge intensive tasks within the academic community and the way these tasks are related to the usage and exchange of documents. The researchers that are going to participate in the study will be asked to point out documents that are related to tasks that they have performed, like the writing of a paper, and also the way that these documents (or some of them) were exchanged during the task. As a result we will hopefully be able to draw a raw map of basic knowledge intensive task and their corresponding document life cycle. We will thus identify the tasks for which the proposed KM services can prove more useful and also construct scenarios of usage. Furthermore we can associate the tasks with the type of application used (e.g. browser, email client) and the corresponding document types (e.g. html documents, emails) and hence guide the design decisions related to the user interface design. Finally it is important to distinguish between different ways in which the exchange of documents is currently performed. As already mentioned the use of a central repository or of a central brokering service is fundamental in the way the proposed architecture supports the exchange of documents. It would thus be interesting to find out how this approach is related to the current document exchange practices. How for example the use of a central repository is affected by privacy issues or by the formulation of collaborative groups within the organisation.

Monitoring of changes in the information needs. The above method does not involve the analysis of the actual content of the used documents. It is important however to track the way the information needs of the users change over time. To do so, we first of all need a way to keep track of documents that have proven useful. We could for example ask the participants to store in a certain folder (one for each one of the participants) every document that they found useful. Even better we could provide the participants with a simple utility that does exactly the above in favour of the user. The user could use the utility's only button to notify that the current document is useful. The utility is then responsible of storing the current document file in an appropriate folder. Having collected the documents that have proven useful for each one of the participating researchers and for a substantial amount of time, we can then compare their content in the order that they have been stored to find out the way the information needs change over time. Of course this implies that we should be able to analyse the content of a diversity of document types like word documents, ps documents, pdf documents, html documents, etc. Although this is not a trivial task we hope that we will find existing publicly available applications that will do the job for us. Whatever the solution to the problem of accessing a document's content, the comparison between the documents' content will reveal trends in the way an individual's information needs change over time. We expect that the identified changes will be relative smooth or that there will be periods of relative stable needs followed by sudden radical changes that are implied by a change of context. In the second case it will be interesting to find out what has caused the sudden change by asking the corresponding user. In the case that the study will reveal that the information needs of the participants change radically quite frequently then the solution is to make the profile to take into account the parameters that cause these changes. If for example the parameter that mostly causes the sudden changes is the task that the individual is performing at any point in time then we could use different profiles for each one of the basic tasks that a user is involved in.

Unstructured Interviews. Finally we can complete the requirements of the KM front-end by directly asking the participating researchers for certain preferences that they have. Although our goal is not to create a system customised to the needs of a certain organisation, it is important to acquire a clear picture of the way knowledge workers work. The goal is to create a pilot KM system that will be easily adopted by KMi's research community so that we will quickly move into testing the system in a real situation. The interviews can provide the extra details that we will need by then and also investigate the ground that the first prototype will be applied. We do not know at this point what this details will exactly be, but we are sure the development process will produce a number of issues that we will have to overcome.

The results of the above studies will feed the development of the KM front-end and although we describe this process as the fifth level of development we believe that it is important to start conducting these studies right from the beginning of the first year and in parallel to the development of the actual profile (see Appendix A'). After a prototype of the actual KM front-end has been developed experiments on real users can be conducted. The goal of the experiments will be to research a number of user-related issues. These issues include:

- *User acceptance of the initial system.* Do users find the initial system performance attractive enough to keep on using the system? Are users willing to provide the system with the necessary feedback that is needed for improved performance? What is the system's usage curve? Does the system's usage falls after a first period of system exploration by the users?
- *Characteristics of the user information needs and interests.* As mentioned in (Nanas, N. 2001), we expect that each one of the users corresponds to a distinct class of information interests. Although some overlapping at the higher level is expected due to the common organisational context, the actual interests are different for each one of the users. We can investigate this expectation in a number of ways. First of all we can evaluate the terms in the *concept layer* of the users' profiles to identify how well they correspond to the actual topic categories of interest in the organisation as a whole. Furthermore the comparison between the *concept layer* of the different profiles can reveal the degree of overlapping of user interests at a higher level. The actual similarity of interests however can be assessed either by clustering the profiles or by finding for each one of the users the best matching profiles of the rest of the users. Finally, in terms of the Planet news server, we can compare the proposed profile's representation of user interests to the existing classification categories of the planet stories. We expect that the results will reveal that a user's information needs and interests can not be easily described in terms of a number of predefined categories.
- *Popularity of services.* One interesting aspect of the proposed system is the interweaving between supported services. It would be interesting to find out which of the supported services are more used. Do users prefer to search for document's themselves or they just wait for the system to actively provide them with information? How often do they use a document to initialise a *document-based* search or to find the most appropriate expert to ask for more explanations? Does the system trigger the communication and hopefully the collaboration between users? The system's log can be used during the experiments to keep track of the user's interaction with the system.

- *System performance.* Of course the most crucial requirement is for the system to provide the user with relevant enough and potentially useful information. Since each one of the users has a different behaviour in terms of giving feedback to the system, the performance of the system can be implicitly measured based on the system's frequency of usage. We can count how often each one of the users uses the system to look for useful information or how often she/he evaluates positively the information that the system presents to him. How much does each interaction with the system long? Do users keep on using the system after the first results are presented, to search for more relevant information? In general, is the system's performance satisfactory enough in order for the users to adopt the system as their provider of useful information?

5. CONCLUSIONS

Our research goal is to investigate the potential use of Information Filtering technology as the core of a Knowledge Management application. The reasons that have pointed towards this research direction are explained in (Nanas, N. 2001). The present document describes the services that the proposed KM system will be able to provide. The Knowledge Sharing Environment (KSE) is used as a prototype and a proof of concept for the proposed services.

At the core of the envisioned KM system user profiling technology is used to represent the information needs of an individual working on a knowledge intensive task. An architecture is described that has the ability to both adapt to sudden modest changes in the user's information needs and to evolve in order to reflect long term radical changes that are implied by a change in the user's context (e.g. change of project). We then present in detail how the proposed KM services are realised based on the profile architecture.

We distinguish between five stages of development. The first four deal with the implementation of the profile's architecture. At each stage we progressively test the effect that the term weighting method, the representation of dependence between terms, the profile adaptation and profile evolution have on the system's performance. However to test the system in a real situation a KM front end has to be built for the IF core. We will have to identify the requirements of this front end based on a user requirement study. The ability of the user's to interact with the system will enable the conduction of a number of experiments for researching the human related issues.

So far in our research we have worked on the first of the five development stages. The results are presented in our pilot study. We have started working on the second stage and planning the user requirements study that will guide the design of the KM front end. A rough timetable of our future research steps in presented in Appendix A. We wish that the proposed system will be as feasible as it sounds and that it will stand to its performance expectations.

6. BIBLIOGRAPHY

Abecker, A., Bernardi, A. and Sintek, M. (1999). **Enterprise Information Infrastructures for Active, Context-Sensitive Knowledge Delivery.** *ECIS'99-The 7th European Conference on Information Systems,* Copenhager, Denmark

Abecker, A., Bernardi, A. and Sintek, M. (2000). **Proactive Knowledge Delivery for Enterprise Knowledge Management.** *Learning Software Organizations - Methodology and Applications.*, Guenther Ruhe and F. Bomarius, Springer-Verlag.

Baclace, P. E. (1991). **Personal Information Intake Filtering**. *Bellcore Information Filtering Workshop*

Baclace, P. E. (1992). "**Competitive Agents for Information Filtering.**" *Communications of the ACM,* 35(12): 50.

Belew, R. K. (1989). "**Adaptive Information Retrieval: using a connectionist representation to retrieve and learn about documents.**" *ACM.*

Borghoff, U. M. and Pareschi, R. (1998). **Information Technology for Knowledge Management**, Springer Verlag.

Church, K. W. (1995). **One Term or Two?** *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.,* Seattle, WA USA, 310-318.

Cohen, W. W. and Singer, Y. (1996). **Context Sensitive Learning Methods for Text Categorization.** *Proceedings of the 19th Annual ACM / SIGIR Conference on Research and Development in Information Retrieval.,* 307-315.

Davies, N. J., Stewart, R. S. and Weeks, R. (1998). "**Knowledge Sharing Agents over the World Wide Web.**" *British Telecom Technology Journal,* 16(3): 104-109.

Domingue, J. and Motta, E. (2000). "**PlanetOnto: From News Publishing to Integrated Knowledge Management Support.**" *Intelligent Systems,* 15(3): 26-33.

Foltz, P. W. and Dumais, S. T. (1992). "**Personalized Information Delivery: An analysis of Information Filtering Methods.**" *Communications of the ACM,* 35(12): 51-60.

Greiff, W. R. (1998). **A Theory of Term Weighting Based on Exploratory Data Analysis**. *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, Melbourne Australia, 11-19.

Kim, J., Oard, D. W. and Romanik, K. (2001). **User Modeling for Information Access Based on Implicit Feedback.** *3rd Symposium of ISKO-France, Information Filtering and Automatic Summarisation in networds,* University of Paris 10, Paris, France

Krulwich, B. and Burkey, C. (1997). "**The InfoFinder Agent: Learning User Interests through Heuristic Phrase Extraction.**" *IEEE Expert*: 22-27.

Krulwitch, B. (1995). **Learning Document Category Descriptions through the Extraction of Semantically Significant Phrases.** *IJCAI Workshop on Data Engineering for Inductive Learning.*

Mc Elligott, M. and Sorensen, H. (1993). "**An Emergent Approach to Information Filtering.**" *UCC Computer Science Journal,* 1(4).

Mc Elligott, M. and Sorensen, H. (1994). **An Evolutionary Connectionist Approach to Personal Information Filtering.** *Neural Networks Conference '94*, University College Dublin, Ireland

Menczer, F. and Monge, A. E. (1999). **Scalable Web Search by Adaptive Online Agents: An InfoSpiders Case Study.** *Intelligent Information Agents: Agent-Based Information Discovery and Management on the Internet.*, M. Klusch, Springer-Verlag**:** 323-347.

Moukas, A. and Maes, P. (1998). "**Amalthaea: An Evolving Multi-Agent Information Filtering and Discovery System for the WWW.**" *Autonomous Agents and Multi-Agent Systems.*, 1(1): 59-88.

Moukas, A., Zacharia, G. and Maes, P. (1999). **Amalthaea and Histos: MultiAgent Systems for WWW Sites and Reputation Recommendations.** *Intelligent Information Agents: Agent-Based Information Discovery and Management on the Internet.*, M. Klusch, Springer-Verlag**:** 293-322.

Nanas, N. (2001). **Literature Review: Information Filtering for Knowledge Management.** Technical Report, KMI-TR-113, Knowledge Media Institute, The Open University.

Nanas, N. (2001). **Pilot Study: Experiments during the first stage of development.** Progress Report, Knowledge Media Institute, The Open University.

O'Leary, D. E. (1998). "**Knowledge Management Systems: Converting and Connecting.**" *IEEE Intelligent Systems*: 30-33.

O'Riordan, A. and Sorensen, H. (1995). **An Intelligent Agent for High-Precision Text Filtering.** *Conference on Information and Knowledge Management (CIKM'95).*, Baltimore, MD USA, 167-174.

Reeves, B. and Shipman, F. (1992). **Supporting Communication between Designers with Artifact-Centered Evolving Information Spaces**. *Computer Supported Cooperative Work*, Toronto, Canada, 394-401.

Rhodes, B. J. and Starner, T. (1996). **Remembrance Agent: A continuosly running information retrieval system.** *First International Conference on the Practical Applications of Intelligent Agents and Multi Agent Technology (PAAM '96),* 487-495.

Sheth, B. D. (1994). "**A Learning Approach to Personalized Information Filtering.**" *Department of Electrical Engineering*: 54.

Sorensen, H. and Mc Elligott, M. (1995). **An Online News Agent**. *BCS Intelligent Agents Workshop, British Computer Society*, Britain

Sorensen, H., O' Riordan, A. and O' Riordan, C. (1997). "**Profiling with the INFOrmer Text Filtering Agent.**" *Journal of Universal Computer Science,* 3(8): 988-1006.

Tauritz, D. R., Kok, J. N. and Sprinkhuizen-Kuyper, I. G. (1999). "**Adaptive Information Filtering using Evolutionary Computation.**" .

Yang, Y. and Pedersen, J. O. (1997). **A Comparative Study on Feature Selection in Text Categorization.** *Proceedings of the Fourteenth International Conference on Machine Learning (ICML '97)*