KnowCat: A Web Application for Knowledge Organization

Xavier Alamán and Ruth Cobos

Departamento de Ingeniería Informática, Universidad Autónoma de Madrid, Cantoblanco, 28049 Madrid, Spain

Abstract. In this paper we present KnowCat, a non-supervised and distributed system for structuring knowledge. KnowCat stands for “Knowledge Catalyst” and its purpose is enabling the crystallization of collective knowledge as the result of user interactions. When new knowledge is added to the Web, KnowCat assigns to it a low degree of crystallization: the knowledge is fluid. When knowledge is used, it may achieve higher or lower crystallization degrees, depending on the patterns of its usage. If some piece of knowledge is not consulted or is judged to be poor by the people who have consulted it, this knowledge will not achieve higher crystallization degrees and will eventually disappear. If some piece of knowledge is frequently consulted and is judged as appropriate by the people who have consulted it, this knowledge will crystallize and will be highlighted as more relevant.

1 What Kind Of Knowledge Are We Dealing With?

Knowledge is a heavily overloaded word. It means different things for different people. Therefore we think that we need to make clear what definition of knowledge we are using in the context of KnowCat.

One possible definition of knowledge is given in the Collins dictionary: “the facts or experiences known by a person or group of people”. This definition, however, is too general and needs some refinement to become operational. In fact, we need some classification of the different types of knowledge that exist. Probably, each of them will require different treatments and different tools to be supported by a computer system.

Quinn, Andersen, and Findelstein [5] classify knowledge in four levels. These levels are cognitive knowledge (know-what), advanced skills (know-how), systems understanding (know-why), and self-motivated creativity (care-why). From this point of view, KnowCat is a tool mainly involved with the first level of knowledge (know-what) and secondarily related with the second level of knowledge (know-how). KnowCat tries to build a repository (a group memory) that contains the consensus on the know-what of the group (and perhaps some indications on the know-how).

According with Polanyi [4] a meaningful classification of knowledge types may be constructed in terms of its grade of “explicitness”: we may distinguish
between tacit knowledge and explicit knowledge. Tacit knowledge resides in people and is difficult to formalize. On the other hand explicit knowledge can be transmitted from one person to other through documents, images and other elements. Therefore, the possibility of formalization is the central attribute in this classification. KnowCat deals with explicit knowledge: the atomic knowledge elements are Web documents.

Allee [1] proposes a classification of knowledge in several levels: data, information, knowledge, meaning, philosophy, wisdom and union. Lower levels are more related to external data, while higher levels are more related to people, their beliefs and values. KnowCat deals with the lower half of these levels, although it may be used to support some of the higher levels.

Finally, there is an active discussion on whether knowledge is an object or a process. Knowledge could be considered a union of the things that have been learned, so knowledge could be compared to an object. Or knowledge could be considered the process of sharing, creating, adapting, learning and communicating. KnowCat deals with both aspects of knowledge, but considers that "knowledge as a process" is just a means for accomplishing the goal of crystallizing some "knowledge as an object".

From the above discussion, we can summarize the following characteristics about the type of knowledge KnowCat deals with:

- KnowCat deals with "explicit" knowledge.
- Knowledge consists in facts (know-what) and secondarily in processes (know-how).
- These facts are produced as "knowledge-as-an-object" results by means of "knowledge-as-a-process" activities.
- KnowCat deals with stable knowledge: the group memory is incrementally constructed and crystallized by use. The intrinsic dynamics of the underlying knowledge should be several orders of magnitude slower than the dynamics of the crystallization mechanisms. KnowCat knowledge may be compared to the knowledge currently stored in encyclopedias. The main difference of the results of KnowCat with respect to a specialized encyclopedia is the distributed and non-supervised mechanism used to construct it. In particular, KnowCat allows the coexistence of knowledge in several grades of crystallization.
- KnowCat knowledge is heavily structured: it takes the form of a tree.

Getting into the details, KnowCat stores knowledge in the form of a knowledge tree. Each node in the tree corresponds to a topic, and in any moment there are several "articles" competing for being considered the established description of the topic. Each topic is recursively partitioned into other topics. Knowledge is contained in the description of the topics (associated with a crystallization degree) and in the structure of the tree (the structure in itself contains explicit knowledge about the relationships among topics).
2 Knowledge Crystallization

KnowCat is a groupware tool [2] with the main goal of enabling and encouraging the crystallization of knowledge from the Web. But what do we mean by crystallization? We conceptualize the Web as a repository where hundreds of people publish every-day pieces of information and knowledge that they believe to be of interest for the broader community. This knowledge may have a short span of life (e.g. a weather forecast), or it may be useful for a longer time period. In either case, recently published knowledge is in a “fluid” state: it may change or disappear very quickly. We would expect that, after some period of time, part of this knowledge ceases to be useful and is eliminated from the Web, part of this knowledge evolves and is converted into new pieces of knowledge, and part of this knowledge achieves stability and is recognized by the community as “established” knowledge. Unfortunately, established knowledge is not the typical case.

KnowCat tries to enrich the Web by contributing the notion of a “crystallization degree” of knowledge. When new knowledge is added to the Web, KnowCat assigns to it a low degree of crystallization; i.e. new knowledge is fluid. When a piece of knowledge is used, it may achieve a higher or lower crystallization degree, depending on the patterns of its usage. If some piece of knowledge is not consulted or is judged to be poor by the people who have consulted it, this knowledge will not achieve a higher crystallization degree and will eventually disappear. If some piece of knowledge is frequently consulted and is judged as appropriate by the people who have consulted it, this knowledge will crystallize; it will be highlighted as relevant and it will not disappear easily. In each moment there will be a mixture of fluid and crystallized pieces of knowledge in the system, the former striving for recognition, the latter being available as “established” knowledge.

Knowledge crystallization is a function of time, use, and opinions of users. If some piece of knowledge survives a long time, is broadly used, or receives favorable opinions from its users, then its crystallization degree is promoted. However, as important as the number of users or their opinions is the “quality” of these users. We would like to give more credibility to the opinions of experts than to the opinions of occasional users. But how can the system distinguish between experts and novices?

KnowCat tries to establish such categories by the same means that the scientific community establishes its members’ credibility: taking into account past contributions. Only accredited “experts” in a given topic can vote for or against a new contribution. To get an “expert” credential at least one of a user’s contributions must be accepted by the already-established experts in the community. In a sense, this mechanism is similar to the peer review mechanism that is widely accepted in academia: senior scientists are requested to judge new contributions to the discipline, and the way for people to achieve seniority is to have their own contributions accepted.

This mechanism is closely related to one of the central aspects of KnowCat: the management of “virtual communities” [3]. But before discussing the details
of such management, we need to describe the KnowCat knowledge structure. KnowCat is implemented as a Web server. Each KnowCat server represents a main topic, which we call the root topic of the server. This root topic is the root node of a knowledge tree. KnowCat maintains a meta-structure of knowledge that is essentially a classification tree composed of nodes. Each node represents a topic, and contains two items:

- A set of mutually alternative descriptions of the topic: a set of addresses of Web documents with such descriptions.
- A refinement of the topic: a list of other KnowCat nodes. These nodes can be considered the subjects or refinement topics of the current topic. The list shapes the structure of the knowledge tree recursively.

For example, we may have a root node with the topic “Uncertain reasoning”. We may have four or five “articles” giving an introduction to this discipline. Then we may have a list of four “subjects” for this topic: “Bayesian Networks”, “Fuzzy Logic”, “Certainty Factors” and “Dempster-Shafer theory”. These subjects in turn are nodes of the knowledge tree, with associated articles and further refinements in terms of other subjects. Among the articles that describe the node (topic), some are crystallized (with different degrees), others are fluid (just arrived) and others are in the process of being discarded. All of them are in any case competing with each other for being considered as the “established” paper on the topic.

Virtual communities of experts are constructed in terms of this knowledge tree. For each node (topic), the community of experts in this topic is composed of the authors of the crystallized papers on the topic, on the parent of the topic, on any of the children of the topic, or on any of the siblings of the topic. There is a virtual community for each node of the tree, and any successful author usually belongs to several related communities.

The mechanism of knowledge crystallization is based in these virtual communities. When one of your contributions crystallizes, you receive a certain amount of votes that you may apply for the crystallization of other articles (of other authors) in the virtual community where your crystallized paper is located. In turn, your contributions crystallize due to the support of the votes of other members of the community. Although peer votes are the most important factor for crystallization, other factors are also taken into account, such as the span of time that the document has survived, and the amount of “consultations” it has received. If a document does not crystallize, after some time it is removed from the node.

The other aspect of knowledge crystallization is the evolution of a tree’s structure. Any member of a virtual community may propose to add a new subject to a topic, to remove a subject from a topic, or to move a subject from one topic to another topic. Once proposed, a quorum of positive votes from other members of the community is needed for the change in the structure being accepted. All community members may vote (positively or negatively) without expending any of their acquired votes. KnowCat supports some elemental group communication
protocols to allow discussion on the adequacy of the proposed change, in case it is needed.

The two mechanisms above described for crystallization of topic contents and tree structure apply in the case of mature and active virtual communities. However, virtual communities behave in a different way when they are just beginning, and also (possibly) in their last days. KnowCut proposes a maturation process that involves several phases. Rules for crystallization (and other rules) may be different for each phase. Figure 1 shows this evolution.

![Diagram showing the lifecycle of a node in a virtual community]

Fig. 1. Virtual community nodes

Normal crystallization mechanisms apply to active nodes (and associated virtual communities). However, these mechanisms require a minimum amount of activity in the node to obtain reasonable results. This minimum might not be achievable at the beginning or at the end of a node’s life cycle.
When a node is created (especially when it is created as the root node of a new KnowCat server), there may not be many accredited experts to form the virtual community. For some time (that in the case of new nodes hanging from established knowledge trees may be elapsed to zero) the node will work in the supervised mode. During this supervised phase there will be a steering committee in charge of many of the decisions that will be made in a distributed way in later phases. In particular, all members of the steering committee are considered experts on the node and have an infinite amount of votes to expend on accepting or rejecting contributions. Radical changes in the structure of subjects are possible by consensus of the steering committee. In addition, one important task of the steering committee is to motivate the community of a new node to participate and to achieve enough size to become an active community. Members of the steering committee are defined when a new node is created. New members can be added by consensus of current members.

A steering committee may decide to advance a node to active status. At this point the committee is dissolved and crystallization of the knowledge is carried out as explained before. In exceptional cases, a community of experts may decide (and vote) to return to supervised mode. This probably will be motivated by the need to make radical changes in the structure of subjects. Active communities can only change subjects by adding or deleting subjects one at a time. Supervised communities may engage in more complex and global structural changes.

Finally, an active community may reach the stable phase. Many of the community members are no longer active, so different rules should be applied to ensure some continuity of the crystalization. Changes are rare, and most of the activity is consultation. Few new contributions arrive, and they will have much more difficulty to crystalize than in the active phase. However, if activity raises to a minimum, the node may switch to active status again and engage in a new crystallization phase.

3 Implementation

We currently have an operative prototype of the KnowCat system, which is a Web-based client-server application. Each KnowCat server can be considered to be the root of a tree structure representing knowledge. This structure and all its associated data (contributions, authors, votes, etc.) are stored in a database.

Knowledge contributions (documents) are not a part of KnowCat. Generating new knowledge (for example, a paper about "Security in Windows NT") is a process that is not managed by the tool. KnowCat presumes that any contribution is a WWW (World Wide Web) document located somewhere in the Web, accessible via a standard URL. The KnowCat knowledge tree stores URL references to such documents.

Users connecting to a KnowCat node can choose between several operations: adding a document, voting on documents, proposing subjects of a topic, and voting on subjects. KnowCat servers build Web pages in a quasi-dynamic way. When an operation is realized, some data is added to database tables but new
crystallization values of system elements are not calculated at this time. Crystallization is performed periodically and asynchronously. KnowCat servers build Web pages dynamically, using information stored in the database.

We now present some of the screens a user would see while working with KnowCat. Figure 2 shows a sample screen shot.

![Screenshot of KnowCat](image)

**Fig. 2.** Root node of a KnowCat knowledge tree

Figure 2 shows the root node of a knowledge tree that was produced in one of our experiments (see Section 4). The screen is divided in two parts. The left side shows documents that have been submitted on the topic of "Uncertain Reasoning". An author name and timestamp identify these contributions. The first two documents in Figure 2 have already crystallized, but the third document is still fluid. Crystalized documents are ordered by crystallization grade.

To visit a document (for example “Rosa Martin Salas [3/23/99 - 5:59:41 PM]”), the user clicks on its link and the corresponding document content (located somewhere in the Web) is displayed.

The right side of the screen in Figure 2 shows the refinement of the "Uncertain Reasoning" topic. Below the arrow ("Next Subjects") are subjects that refine the current topic. An associated knowledge node may be visited by clicking on the desired subject. For example, if we select the subject “Methods based on Fuzzy Sets Theory”, the screen shown in Figure 3 would be displayed.
In the lower-right corner of Figure 3 we see that a “Fuzzy Measures” subject has been proposed for adding to the refinement list of the current node. By selecting the option “TO VOTE... ADD SUBJECT”, any authorized user (a member in the steering committee of a supervised node or a member of the virtual community otherwise) may contribute to this proposed modification being accepted. If someone proposes to remove a subject, it will appear below the previous section under the heading “Subjects Proposed to be Removed.”

Fig. 3. Refinement node visited from the root node

In any node, authorized users may use the options “VOTE DOCUMENT” to vote (positively or negatively) on one of the documents of the node. Any user may contribute to the system by means of the “ADD DOCUMENT” option.

4 Experimental Results

We have performed three experiments with KnowCat. The first experiment was done with third-year students enrolled in an advanced operating systems course in the Computer Science Department of the Universidad Autónoma de Madrid.

At the beginning, a KnowCat node was created on the topic “Operating Systems.” An initial structure of six subjects was devised. We wanted to check
the "contents" part of the tool. Collaboration was voluntary, and only 15 of the 80 students enrolled in the course participated. Of those 15, 7 contributed regularly.

Students performed several operations:

- 27% of operations were adding documents to the system,
- 23% were voting on documents,
- 14% were proposing initial-topic subjects, and
- 36% were voting on proposed refinements.

This experiment served as a test bed for some of our initial ideas, and several of the developments presented in this paper were devised after our initial findings. Not surprisingly, one of the major problems we faced was to motivate the participation of the students (contributing and voting). The results of the interaction were of moderate quality in this first experiment.

The second experiment involved students of a graduate Computer Science course on "Uncertain Reasoning" at our university. For this experiment we wanted to test the "structure" part of KnowCat, so we created a root node ("Uncertain Reasoning") with no initial topics. This course is delivered every year and we will continue this experiment for several more years. About 10 to 15 graduate students enroll each year. The aim of the experiment during the first year (this year) was to check the feasibility of the students making a good structure for the topic by using our proposed voting mechanism.

At this point we can summarize the experiment progress:

- Most of the operations performed by students were related to the refinement of topics. 41% of these operations consisted of proposing new subjects; 30% were student opinions on new subjects; only 13% consisted of adding documents to the system; and the remaining 7% consisted of votes on documents.
- There are 14 topics in the KnowCat node and they are distributed over a tree four levels deep. So the structure of the node has evolved very well. The course instructor judges the quality of the structure to be satisfactory. He did not interfere directly in the process of defining the structure.
- Student participation has been fairly uniformly distributed in time. However, there were several periods during which student participation was markedly greater. These coincided with student communication by e-mail. For example, when students noticed that they should discuss the correct location of a refinement, they used e-mail communication for the discussion.
- During the quarter (3 months) each student participated 10 times on average, achieving a much higher degree of participation than in the previous experiment. The graduate status of the students in the second experiment, as well as the more controversial nature of the process of deciding the most adequate structure of a topic, are likely reasons for the higher participation.
- Students spent a lot of time consulting documents contributed by other students. This suggests that the system is really useful as a repository of structured knowledge.
Another interesting design aspect has arisen as a result of this experiment. In the current version of KnowCat, when a user proposes a new subject for a topic s/he only has to specify the new topic’s name. Sometimes just the name is not significant enough to understand the intent of the proposal, and therefore discussion on the adequacy of the proposal is not easy. This issue will be solved in future versions of the tool by allowing the inclusion of commentaries with the initial proposal and subsequent votes.

Finally, the third experiment involved students at our university registered in a preliminary operating systems course. This experiment started with the root node created in the first experiment. The 200 students (two concurrent classes) were assigned to 12 predefined subjects (16 students per topic). Each student was assigned to produce a small paper on the assigned topic and vote for the three best papers in that same topic. We wanted to check the hypothesis that when you get enough contributions and enough votes from “knowledgeable” peers, the result is a reasonable description of the topic. The instructor graded papers independently, and this grading was used to check the adequacy of the voting system to capture the quality of the papers. In this case student motivation was achieved by grading the students on the quality of their papers and in the quality of their votes (that is, in their judgement capabilities on the topic, as compared with the opinion of the teacher).

The results of this experiment were very encouraging. In 11 out of the 12 topics the votes of the students converged to a small set of papers. There was a remarkable consensus. For most topics the two most popular papers collected 50% of the total votes (the maximum was 68% since each student issued three votes for three different papers). Also, the four most popular papers always collected 75% of the total votes. Only in one topic was there a dispersion in the votes, probably due to a more uniform distribution of the quality of papers submitted.

Furthermore, in 10 out of the 12 topics at least two of the three papers selected by the instructor as “the three best papers” were also selected by the students as such (in 3 cases the two sets of “the three best papers” were identical). In 7 topics the best paper, in the opinion of the instructor, was selected by the students as one of the three best. There was one topic where the opinion of the instructor was very different from the opinion of the students, but this case was also the one where the votes were most dispersed: there were no clear “winners.”

Although these results have to be verified with further experiments (next year this experiment will be repeated with other students), we think that they already provide some support for two of the hypotheses that underlie the design of KnowCat:

- If a set of “knowledgeable” people engage in a reasonable interaction with our system, the result converges to some consensus.
- This consensus is closely related to some objective measure of “quality” of the contributions.

Of course, these results have to be pondered in the light of our experimental setting, which has possibly introduced some artifacts:
Motivation aspects were intentionally omitted from the experiment. Grading policies furnished sufficient external motivation to the students. In particular they were encouraged to vote after a careful reflection on the quality of all the papers. It is not clear whether the obtained consensus would have been achieved without this external motivation.

The group of “knowledgeable” people was very special in nature. All were novices in the topic at the beginning of the course, and they acquired their expertise by attending the same lectures and by reading the same reasonably sized but finite bibliography.

The “objective” measurement of the quality of the papers was done by the same person that had lectured them during the course.

During the next year we are going to engage in further experiments trying to correct these artifacts by changing the experiment conditions.

5 Conclusions And Future Work

In this paper we present a new tool for structuring knowledge in the Web based on the concept of “crystallization.” A first prototype of the tool is currently in operation. Some of the findings in our initial experiments have been included as new features of the tool. However, we have identified several areas where further improvement and experimentation are needed.

KnowCat will be a distributed and scalable system. It should work with stand-alone servers (“knowledge islands”) and also with combinations of these “islands” into higher level structures. Inter-server protocols must be developed for this purpose. Several new problems appear in this scenario, such as the possible duplication of nodes (perhaps with slightly different names), problems with the ownership of the joint structure, and difficulties in structure changes that affect two servers simultaneously.

We currently propose a tree structure for the knowledge tree. However, it may be the case that a subject is the refinement of more than one topic. Perhaps the tree structure should be transformed into a generic graph structure. The compromise is between efficiency, clarity, and maintainability (tree structure) and expressiveness (graph structure).

There are several open questions with respect the crystallization mechanism. For example, some enforcement of the “fairness” of the voting mechanism is necessary. The votes related to such crystallization procedures will be public. Anyone will be able to research the voting structure and locate recurrent “voting cycles” or other artifacts. The dynamics of the voting mechanism also presents several open research issues. Should the system be inflationary on the number of votes available? When an article de-crystallizes, what should be done with the author’s votes obtained through the article for him?

At the moment we have only tested KnowCat on university courses, and we have noticed that this system is really useful for motivating students in sharing their knowledge and incrementally constructing a knowledge repository that will
improve over the years. However, we expect our tool to show some of its more innovative characteristics in the environment of research groups interested in sharing knowledge. During the next year several experiments will be carried out in this direction.

Acknowledgements. This work has been partially supported by the Comunidad Autónoma de Madrid (project number 07T/0027/1998) and by CICYT (project number TIC98-0247-C02-02).

References
