Quality of Service in Knowledge Collection and Management

Eric Hughes, Daryl Morey, Arnon Rosenthal
The MITRE Corporation
hughes,dmorey,arnie@mitre.org

Abstract

This position paper proposes new approaches that are intended to allow an organization to collect and manage knowledge while providing flexible quality of service mechanisms for access to the knowledge. We consider both the means of managing the knowledge in databases and of collecting knowledge from users.

1 Introduction

An organization's ability to manage knowledge depends on a successful knowledge service (KS), which must support the collection of knowledge from users, make knowledge available to users if it exists, provide knowledge that is effective in the work of the organization, and do all of these with appropriate quality of service. This position paper gives the rudiments of an ambitious proposal for such a KS, by utilizing existing approaches from database, artificial intelligence, and from new approaches for knowledge collection and management. We envision a KS that is utilized by a variety of users and applications and includes a variety of knowledge bases (KBs) administered by specialists. Each user and application may have different needs and expectations of the proposed KS, and these needs can be expressed by quality of service (QoS) parameters. For example, QoS parameters for the KS might include level of abstraction and timeliness of delivery of knowledge.

Imagine a KS to be used by medical researchers and practitioners. Epidemiologists would want access to summarized knowledge of treatments and also the ability to correlate details. For advising a single patient, one would want diagnosis and treatment data for others matching the patient's profile. The KS could also track the growth of knowledge over time. A medical care auditor needs to understand this history to determine whether past treatments were appropriate, by the standards of the time.

This position paper proposes new research in incentive systems to get users to supply knowledge, and in ways that explicit description of needed QoS can increase access efficiency and flexibility. For knowledge collection, we plan to develop an incentive model that recognizes the value of intellectual property and the value of an employee's intellectual contribution to an organization. For efficient management, we propose to design a KS that provides object-oriented views of KBs and includes QoS parameters. Our approach is intended to benefit from existing research in database views.

One analogy that we have found useful compares a *meme*, or unit of knowledge, to a gene [Daw76]. Here we strive to treat knowledge in a KS like a meme. One such treatment is to expose a meme to competition, to see how well it compares with other memes. Another is to measure a meme's contribution to the organization by the effect of the new knowledge to which it leads.

2 Providing Knowledge

One of our goals is to motivate users to provide and access the knowledge that will be stored in KBs. Although we think of people as users in this context, similar statements might be made for software agents.

The first problem we face in developing a KS is to make it beneficial to users who are asked to accept the inconvenience of having to provide knowledge.

The copyright of this paper belongs to the papers authors. Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage.

Proceedings of the 4th KRDB Workshop
Athens, Greece, 30-August-1997

(F. Baader, M.A. Jeusfeld, W. Nutt, eds.)

http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-8/

Typically, knowledge which is useful to a wider variety of users requires more effort on the part of the provider (e.g., to categorize or provide additional details). With conventional databases, lack of incentive to the provider is one of the biggest reasons for poor data quality. The growth of the web shows that some people are willing to provide useful information once it is easy for them to do so but most are unwilling to manually maintain the quality of that information. We propose incentive-oriented models for knowledge collection and maintenance.

Incentives need to be tailored to different communities. For organizations such as accounting departments whose *purpose* is the creation of knowledge (or at least data), mandates may suffice. In communities with considerable motivation toward a shared objective (e.g., professional specialties), *prestige* may be sufficient incentive for participation. One tactic that may be effective is to provide tools that help the user with their job, and at the same time build KBs. For example, a text processing system that provides appropriate tags for title, author-list, references (as in bibTex) can help users (e.g., to reformat their papers to the demands of different publications); these tags can then be exploited by search and summarization tools. As another example, users are quite willing to use search engines, and their usage records could help identify individuals with particular expertise or interest. Information providers are motivated to register their presence with the search engines, and sometimes tailor their web pages to increase their apparent relevance to the search engine (leading to counter-measures in the search engines, a scenario familiar from biological competition).

Misuse of collection techniques and incentives must also be considered. Tactics such as use of search records threaten serious breaches of privacy. Some users may fear that sharing their stock of information decreases job security. In many circumstances, the attitude of "why bother" seems a more formidable obstacle. Incentive systems may be manipulated by collusion between providers. This behavior seems especially likely with monetary incentives, but reciprocal invitations to coauthorship are not unknown in prestige-driven communities. We plan to perform an experiment comparing incentive models for publication of knowledge to our intranet.

The collective knowledge of an organization is far from static. Since no quality metric is likely to be appropriate for all kinds of knowledge, we propose a KS that will allow users and applications to provide feedback on how well the supplied QoS parameters were met for a given query. Thus, whenever knowledge is accessed, the consumer will have the opportunity (and, perhaps, obligation) to annotate that knowledge, and those annotations will eventually modify and/or become part of the knowledge available to the KS. A similar approach is used in MIT's Idea Exchange [MIT97], and we plan to extend this to feedback provided by applications and to include the value of feedback in our incentive model. We intend this approach to allow us to migrate our current intranet, with publication of annotated documents, to a large but simple KB in which documents representing memes can compete and be judged on their performance.

Today, employees often gain prestige within an organization by sharing their knowledge only selectively. Our approach significantly affects this organizational model because our incentive model may not coexist easily with existing corporate reward structures. Thus, we recognize that our KS will not be successful without commitment on the part of that organization to reward those who provide valuable knowledge. We hope to mitigate this problem by seeding the KS. We also expect seeding to be needed to make participation worthwhile for the users who publish knowledge.

3 Accessing Knowledge

If we can collect knowledge effectively, we must face the problem of managing the result. This section proposes some requirements and approaches for three facets of KS management: openness, explicit quality of service (QoS) parameters, and the need for externalization.

3.1 Openness

Many government and military applications suffer from being tied to non-commercial technology, which in its time was very sophisticated but has since been supplanted by standard approaches. We will attempt to avoid this fate by designing an open KS, with which new client applications have efficient access to knowledge and can adapt the essential QoS mechanisms of a KS as needed. Services such as mediation will be encapsulated as components, to simplify evolution. We recognize the need to be able to build unanticipated database *views* of the knowledge, and to materialize those views (see below) in certain situations. We plan to investigate the utility of KBs as they grow, and to experiment with alternative organizations.

One dimension of openness is the ability to add new KBs to a KS. In similar work, Su et al. have described the OSAM KBMS and emphasize the need for interoperability of rule-based systems [Su95]. Our goal is to provide for efficient sharing of knowledge between any applications. While we do not anticipate the need for novel techniques for knowledge exchange, we recognize that data exchange between federated databases is far from solved and that a wide variety of data exchange approaches might be applied. We plan to start with very simple KB representations (e.g., documents with metadata attributes).

3.2 Explicit Quality of Service

We suspect that a user who requests information from a KS will often want to control the processing of the query, e.g., by supplying parameters that describe desired tradeoffs. The proposed KS will recognize QoS parameters supplied by users and applications. While this has been partially addressed in the context of approximate query results (see, e.g., [Dog96]), we see a need for a much wider set of QoS parameters, including knowledge quality, timeliness of knowledge, speed of access, level of abstraction, knowledge source(s), and security classification. Each parameter thus defines a *dimension* of service in our proposed KS. Since we cannot afford to build a mechanism to support every possible dimension, we instead propose that the KS support a basic set of dimensions, and allow users and applications to supply the KS with the means to consider new dimensions (e.g., by providing a quality-checking function). Unlike conventional DBMSs, this requires that query optimization (at the KS level) be separated from the underlying KBs. As our KS is scaled up to the size of an intranet, it will become increasingly important to share and calibrate these functions among groups of users. Sophisticated QoS functions and calibration techniques can be found in artificial intelligence research.

A database view can be augmented by its QoS parameters, yielding a KB view. Often these views will be parameterized, e.g., allowing "timeliness". The parameters for customizing both data and QoS portions of a user's KB view will often be derived from the user's profile). Such views may be supported by a combination of push and pull technologies, transparent to the user. For example, for very fast access but higher maintenance costs, one can *materialize* the view by maintaining local copies of view-defined objects on the client (e.g., [Kun96]). This *push* technology is supported by many DBMSs (i.e., view materialization based on either triggers or logs), and seems a useful approach for many KS clients. The general principle is that views describe behavior (i.e., data schema and QoS parameters), and user requirements are compared to these behaviors. The implementation that provides these behaviors is hidden from the clients.

3.3 Concluding Remarks

The use of an effective KS enables an organization to start on the path to becoming a learning organization, which fosters the creation and refinement of memes. With a sufficiently flexible KS, knowledge can be accessed easily and thus can propagate more rapidly through an organization.

We anticipate applications of the proposed KS in the areas of corporate KBs, planning and simulation systems, and digital libraries. We have begun work in knowledge collection by creating a macro for our corporate-standard word processing application, which allows document metadata to be published to a KB and evaluated by the creator or by reviewers. This simple KB will be used in preliminary experiments on knowledge management and on incentive models for knowledge providers. We suspect that these experiments will determine whether such a KS can be constructed with tools like Lotus Notes, or whether commercial object DBMS or KBMS products are more appropriate.

3.3.1 References

References

[Daw76] R. Dawkins. The Selfish Gene. Oxford University Press, 1976.

[Dog96] E. Dogdu and G. Ozsoyoglu. Adaptive Real-Time Transactions and Risk-Based Load Control. Workshop on Databases: Active and Real-Time, 1996.

[Kun96] H. Kuno and E. Rundensteiner. Using Object-Oriented Principles to Optimize Update Propagation to Materialized Views. ICDE, March 1996.

[Kus95] J. Kusch and G. Saake. Supporting Autonomy for Information Systems in a Changing Environment. KRDB 95.

[MIT97] http://learning.mit.edu/ide/.

[Su95] S. Su et al. An Extensible Knowledge Base Management System for Supporting Rule-Based Interoperability among Heterogeneous Systems. CIKM 95, p. 1-10.