Annotate! A Web-based Knowledge Management Support System for Document Collections

Mark Ginsburg
Leonard N. Stern School of Business
New York University

Ajit Kambil
Leonard N. Stern School of Business
New York University

June 1998

Working Paper Series
Stern #IS-98-19

Annotate! A Web-based Knowledge Management Support System for Document Collections

Mark Ginsburg *

Ajit Kambil †

{mginsbur,akambil}@stern.nyu.edu

June 5, 1998

*Doctoral Student, Information Systems Department, Stern School of Business, New York University, 44 W 4 St., 9-181 MEC, NY NY 10012. tel +1 212 998 0835 fax +1 212 995 4228. This paper is available in PDF format at http://raven.stern.nyu.edu/papers/hicss.pdf.

†Assistant Professor, Information Systems Department, Stern School of Business, New York University.

Abstract

Annotate! A Web-based Knowledge Management Support System for Document Collections

Knowledge management is an increasingly important source of competitive advantage for organizations. Knowledge is often a renewable, re-usable and accumulating asset of value to firms that increases in value with employee experience and organizational life. Knowledge embedded in the organization's business processes or the employee's skills are assets are generally hard to discern, accumulate and replicate by competitors. It provides the firm with unique capabilities or "resources" to deliver customers with a product or service. In contrast as we undertake electronic commerce, customer interfaces and business strategies generally become more visible to competitors. Thus the organizations capacity to effectively accumulate and leverage knowledge assets better than its competitors becomes a key source of competitive differentiation.

As firms become more knowledge intensive, more effort is being expended on knowledge management (KM). While much progress has been made on designing IS to support decision making, the art and design of KM systems to preserve, index, formalize and leverage knowledge in organizations is still new (see O'Leary (O'Leary, 1998) for a review of best practices). Knowledge is fundamentally more complex than information or data, and systems supporting knowledge management have a broader range of design issues.

This paper reviews approaches to knowledge management support systems (KMSS) and proposes the need to design systems that carefully map their features to target organizations and user groups. We illustrate `Annotate!` as a specific KMSS designed to support the knowledge management of document collections in federated organizations which lack common vocabularies and central authority.

1 Knowledge Management Support Systems

Knowledge management support systems require new design principles because knowledge fundamentally differs from information and data in organizations. Knowledge is an organizational member's experience and values combined with and shaped by the information contained in various systems and data provided to the person (Davenport and Prusak, 1998) (Nonaka, 1994). It is intrinsic to organizational members and focuses on the information recipient. In contrast, data refers to a set of discrete, objective facts about events recorded in an organization and information provides organizational members with contextual meaning to the data.

Knowledge can be tacit (t) or explicit (e)(Nonaka, 1994). Tacit knowledge is the beliefs and values that are hard to express but inferred from the behaviors of organizational members. Explicit knowledge is easily expressible such as the formalization of an organizational routine or process through a flow diagram. Organizational and individual knowledge is created through a continuous dialogue between the tacit and explicit knowledge of individuals. Ideas are formed in the minds of individuals, but interaction between individuals typically plays a critical role in developing these ideas. Nonaka identifies four processes for individuals to gain knowledge. These processes include: socialization ($t \rightarrow t$), internalization ($e \rightarrow t$), externalization, ($t \rightarrow e$), and combination ($e \rightarrow e$).

While new knowledge is developed by individuals, organizations play a critical

role in articulating and amplifying that knowledge (Davenport and Prusak, 1998). This requires organizations to provide a working infrastructure, composed of a set of knowledge management support systems (KMSS), and meaningful policies for knowledge sharing. Such an infrastructure would allow users to easily share information, with policies that provide incentives to organizational members to participate in knowledge sharing and refinement activities. The information shared among members should reflect their values and beliefs about the information stored and exchanged to support KM.

As KMSS are embedded within an organizational system they must also be designed to fit within the cultural values, authority structures and other design features of the organization. Thus knowledge management consists of both the implementation of information systems and organizational systems with incentives, processes, and tasks to collectively generate, refine and manage organizational knowledge. As IS systems increasingly support KM we denote systems supporting KM as KMSS to note that an information system is only a support tool in an overall organizational KM system.

The ideal knowledge network as conceptualized by Nonaka assumes efficient search and retrieval of an abstract knowledge base; however, he does not indicate design approaches which would bring about this efficiency. This paper introduces the `Annotate!` system to address one segment of this problem, the design of an enhanced retrieval software tool for retrieval on un- or semi-structured document archives. The `Annotate!` system captures user histories

in a typical search session and permits the users to commit annotations which become logically bound to the core documents.

The remainder of the paper reviews critical issues for the design of KMSS in Section 2 and moves onto KMSS challenges, both technical and organizational, in Section 3. Section 4 presents the `Annotate!` system, our software tool to explore KM in the domain of Web-based document archive structure and retrieval. A tour of `Annotate!` is presented from the user's perspective and specific features of interest, such as document annotation and filtering are discussed. Section 5 presents first a technical review of the two fundamental data structures underlying `Annotate!`, the *discussion* and the *session* data. In addition, this section also presents the organizational implications of this architecture. Section 6 discusses briefly an ongoing field experiment with the `Annotate!` system. In Section 7 we discuss lessons learned from this project and provide concluding remarks.

2 Critical Issues in the Design of KMSS

Despite the widespread interest on KM in general, there has been surprisingly little work on what might constitute an effective KMSS and the tradeoffs an organization might face in achieving its KMSS goals. For example, KMSS systems often have some or all of the following components (O'Leary, 1998):

- A Data or Knowledge warehouse. However, as the organization ages and continues to store transaction data in the warehouse, the costs to ensure efficient retrieval on the data store may increase sharply.

- Knowledge search and discovery mechanisms. This problem becomes particularly difficult in the case of multimedia, for example streaming audio and video.

- Knowledge representation via an ontology. There is a significant tradeoff here, too. If an organization imposes an ontology on a series of document collections, there is the possibility of vocabulary conflict across business units. As Pejtersen notes (Pejtersen, 1998), there is a significant cost associated with forming classification schemes which cover the organization's various work domains.

- Knowledge quality control. Establishing a minimum level of credibility for a given knowledge base is an important organizational goal.

- Knowledge visualization techniques. For example, Phelps and Wilensky (Phelps and Wilensky, 1996) have been researching Java applets at the client side to improve the presentation of documents (separating them into text, scanned OCR pages, and other layers).

These components have to be integrated into a system that provides the functionality in the previous sections and maps to organizational requirements.

The integration is done through the human organization and processes that overlay a KMSS. The integration works best when the KMSS features fit well with the organization structure, processes and values.

Without effective retrieval, information islands in a federated organization do not diffuse well across intra-organizational boundaries. Hence, knowledge transfer is limited in any structure with sub-optimal retrieval facilities.

3 Challenges of Designing a Web-based Document KMSS

There are both technical and organizational factors which impact the design of a Web-based document KMSS. In this section, we review the key properties of documents in a Web development environment and discuss key features of the organizational document publishing process that we must keep in mind when designing the KMSS.

3.1 Documents as Web Knowledge Bases

In contrast to well-structured fielded database, unstructured or semi-structured (template-based) documents represent an increasingly important part of organizational knowledge bases. Documents have the potential to be highly expressive, with embedded multimedia objects. While expressive and

strong in presentational markup (rendering) they are often poor in semantic markup making knowledge search and discovery difficult.

The Web facilitates distributed document publishing by virtue of its open HTTP protocol (Baldwin and Clark, 1997), however professional document work products typically incur a high cost of creation in time and effort.

3.2 Document 'Marketing' on the Web

Document repositores which span multiple intranet Web servers pose a marketing problem. With the advent of low-cost WWW publishing, it is quite easy to place a document on a given intranet server. It is quite another matter altogether to to let other business units know that a new document repository exists, or that interesting new documents exist on a server that another business unit may not consult very often.

The Web moves the firm to a peer information model, where clients can easily access servers throughout the intranet. Intranets in federalist organizations (those with semi-autonomous business units) (Ross and Rockart, 1996) face practical difficulties. If each business unit maintains its own intranet server, a given business unit may become used to searching only its own server. How to increase the scope of the search so that functional overlaps between business units might be exploited? Note that the increased scope means that there is greater information throughput (and consequently, greater potential

for knowledge gain) in the aggregate.

3.3 Pre-Coordinate Ontology vs. Post-Coordinate Full Text Search

Document indexing and search can be implemented through pre-coordination or post coordination. In pre-coordination, the documents are associated with subject headers by a collection administrator. The subject headers follow a standard order, for example `Mexico | Economy | Inflation`. Post-coordination is so named because the keywords are combined at search time; there is no subject term taxonomy specified *a priori*.

Pre-coordination implements a centralized ontology, but the effort to set up an ontology and classify documents is manually intensive. As a knowledge base grows, it becomes difficult and expensive to create ontologies and reconcile classifications to suit the interests of many different users. This problem is compounded as the interests of the knowledge/information seekers increase and diverge. Many real KMSS systems which implement static ontologies for classification and selection of control vocabularies face this issue.

If an organization decides to map documents in heterogeneous databases to knowledge structure, as described in the Andersen consulting case (O'Leary, 1998), the maps themselves are susceptible to political processes, often hiding controversial areas and thus limiting the total amount of information available

(Davenport, 1998).

Organizations usually resort to post-coordination or full-text search and impose no vocabulary control. In standard Web-based full text search, we encounter problems such as homonymy, where words mean different things in different contexts, lowering precision and synonymy: search engines that lack a smart thesaurus will artificially deflate the confidence scores of documents containing synonyms to the keywords (Svenonius, 1986).

3.4 Organizational KMSS design challenges

In addition to technical challenges organizations often lack adequate incentives for knowledge sharing and management (cf. Section 5.1.1). These difficulties are often exacerbated in emerging federalist organizations which are dynamic, team based problem solving structures with distributed authority. The first decision business units make is the choice of specific groupware products, such as Notes (Domino) or intranet product suites (Ginsburg and Duliba, 1997); the broader issue is how to organize the documents accessed by the groupware product to facilitate knowledge transfer.

As a result it is not surprising that most systems in the past have covered limited domains (see the O'Leary examples (O'Leary, 1998).) As document publishing is simplified, and Intranets link individuals in organizations to rapidly expanding Web document bases, the previous problems in the design and

maintenance of KMSS become more pronounced. To address some of these problems we have developed Annotate! which provides a flexible KMSS to support federated organizational document management.

4 Annotate! A Web-based Document KMSS

Typical Web Full Text Search (WFTS) engines which provide post-coordinate search have deficiencies which translate into inadequate support for KM. For example, there is no way to share resource discovery made during the course of an ad-hoc search session for one's future use or between users. There are also extremely limited data and metadata clues to assist the user as he or she traverses the system from the front-end (the Query Layer) to the intermediate layer, which is an array of hyperlinks to the core documents (the Retrieval Layer) and on to the bottom layer, the Document Layer. In a typical implementation, the user has no knowledge of others' prior searches or results at the Query front-end and has very few clues of what the most interesting documents might be at the Retrieval layer.

To redress these deficiencies and support a KM platform that is targeted particularly at federated organizations with many document archives (often scattered across multiple Web servers), we have built the Annotate! System[1].

[1] A demonstration version is available on the Internet at http://edgar.stern.nyu.edu/annotate.

One of the driving factors behind the Annotate! design is to enable Nonaka's knowledge management processes of socialization, internalization, externalization and combination by:

- capturing individual and aggregate document appraisals (a means to aggregate individuals' externalization, or use of metaphor to express others' tacit knowledge)
- using individual appraisals of documents to augment document content (to support readers' internalization on an ongoing basis)
- using individual appraisals of documents to support a recommender system (which improves the efficiency of the search by filtering out unwanted documents, for example those from an untrusted domain),
- using individuals' free text annotations to support combination or the reshaping of information and data from one information system to another;
- and using the free text annotations as well to weakly support socialization or the transfer of tacit knowledge from one individual to another.

Annotate! is predicated on the principle that the users and creators of knowledge best know the information relevant to their knowledge management task and that they can more effectively filter, discover and signal useful knowledge to their peers than an automatic system.

As annotations accrue in the system, so do the reasons the annotator had for making the note. Both the annotation text and its rationale are logically bound to the core documents, thus increasing the semantic content of the document repositories[2].

4.1 A Tour of the Annotate! System

In a typical user session, the Query interface as shown in Figure 1 resembles that of a standard Web Full Text Search Engine, for example, Alta Vista, or Excite. The user enters keyword(s) to reach the Retrieval interface, a set of hyperlinks to base documents. There are two enhancements to standard full text search shown in this figure: the first is the ability to filter the result set by annotation domain or to set a minimum aggregate quality rating. The second

[2] Annotate! implements a *star* structure: "for each document, there is only one level of annotations — annotations of annotations are not possible. Stars are simpler for users in some ways because one can read through all unread annotations in a sequence. Since new annotations are always appended to the end of the list, one knows that readers are seeing the same thing, and thus the conversational style of communication is well modeled" (LaLiberte, 1998). The alternative, a *tree* structure where annotations can be made to annotations, diminishes the distinction between the (lengthier) core document and the brief secondary annotation — we wish to make the annotation process simple and limit the length of the annotation entity while keeping attention on the core document which had a greater social cost to produce. However, the tree structure works well where the core document is also a brief note, such as LaLiberte's HyperNews system (hosted at http://www.hypernews.org/) for Web-based threaded Usenet-style discussions.

is the ability to report on the most commonly queries keyword(s) to date, or the most heavily annotated documents to date, or the highest rated documents to date along several dimensions (using session data, cf. Section 5.2).

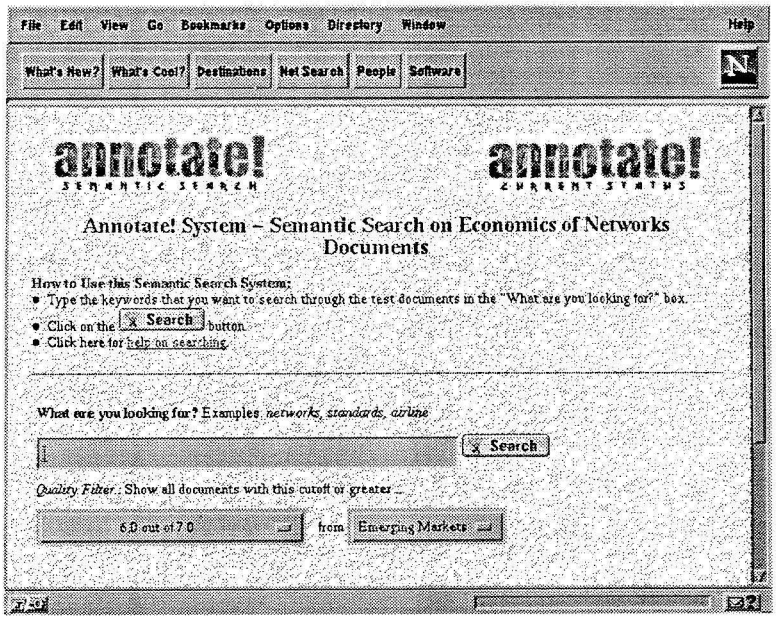

Figure 1: Annotate! Query Interface

The result of the query is an array of hyperlinks, termed the Retrieval layer, as shown in Figure 2.

On the left we see the two most recent annotations; followed by the aggregate Factual Accuracy score (on a scale of 1 to 7) and Quality Score. On the right we have the conventional Excite confidence score followed by a hyperlink to the core document. We display the two most recent annotations in the Retrieval layer and some aggregate statistics about the document annotations to date:

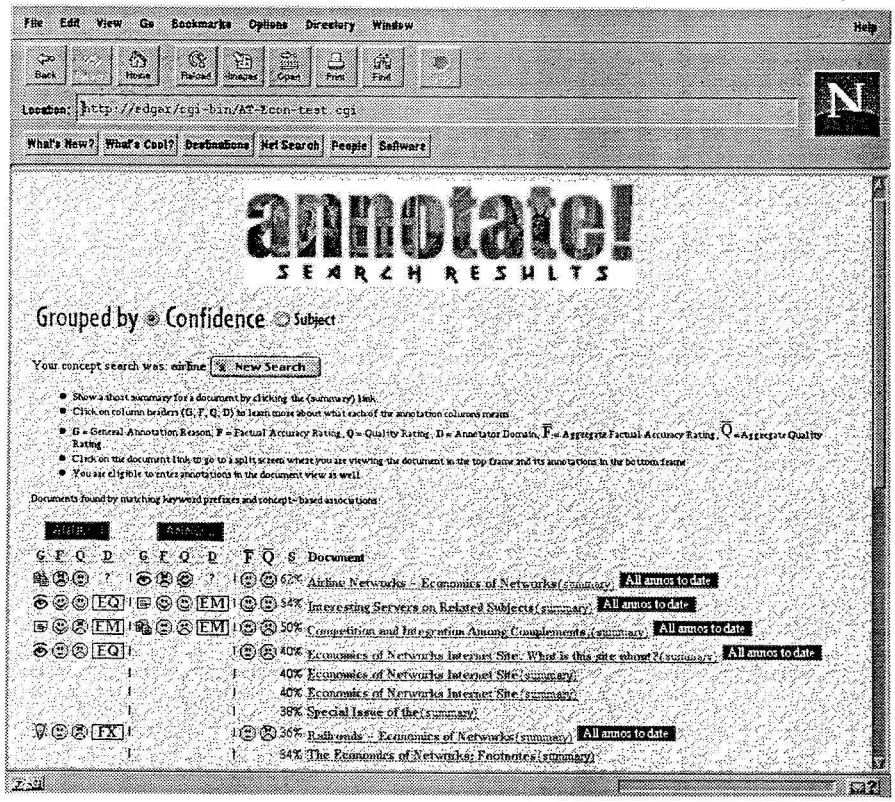

Figure 2: Retrieval Layer Alterations in the Annotate! System

referring to Figure 2 and Figure 3 (detail), the column headings are as follows: G is the general reason for annotating; the light bulb icon represents "a more general idea can be drawn"; the eye icon represents "see also this link ..." and so on. F and Q are user satisfaction measures with the factual accuracy and quality of the document, respectively, on a Likert 1—7 scale. These are mapped to a spectrum of facial expressions similar to Koda and Maes (Koda and Maes, 1996) in their interface agent usability study. $\bar{F}$ is the aggregate factual accuracy rating and $\bar{Q}$ is the aggregate quality rating.

Figure 3 shows a detail of the icons in the Retrieval layer which are created by annotations. The first four icons on the top row represent the most recent annotation for the Document *Airline Networks - Economics of Networks.* In the second row, the left-most 'eye' icon means that the annotation presumably will contain a 'see' link, referencing anothe document for further information[3]. The most recent annotation in the second row originates from a known client IP number that the server is able to map to the *EM* business unit (Emerging Markets); the second most recent annotation in the second row is mapped to *EQ* (Equities).

There are several choices to add clues at the Query layer. With the discussion object, we have implemented recommender filters on the basis of aggregate user appraisals.

[3]An interesting modification of the `Annotate!` system would be to full-text index the 'see' references too; in this way, the document corpus grows as 'see' annotations accumulate.

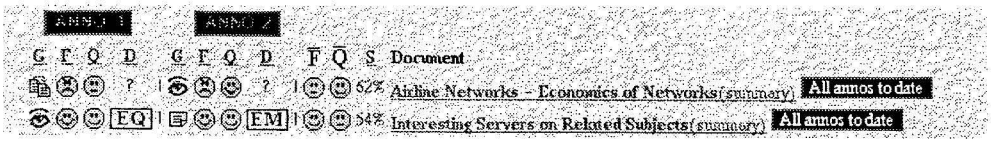

Figure 3: Document Annotations: Detail

The user selects a document hyperlink in Figure 3 to reach the Document Layer.

Figure 4 shows the user in the process of creating a new annotation; the form is kept simple in order to encourage participation in the system. Annotations grow the discussion data store and make add value to the document recommender system. Consider that with no or little underlying appraisals, a filter on aggregate quality would not accomplish the desired effect. However, as users (and by extension, business unit groups) contribute annotations over time, filtering can become a powerful mechanism to limit spurious results. This is depicted schematically in Figure 6.

To accomplish knowledge search and discovery mechanisms, documents and annotations are indexed using the Excite search engine. The user can search using keywords, and refine the search filtering on annotation variables. Knowledge quality control is a subjective process which is completely dependent on the user. Annotations can provide readers with rich data and opinions to aid belief development about the documents. Furthermore, readers who frequently contribute high-quality annotations can become opinion leaders and in a fully

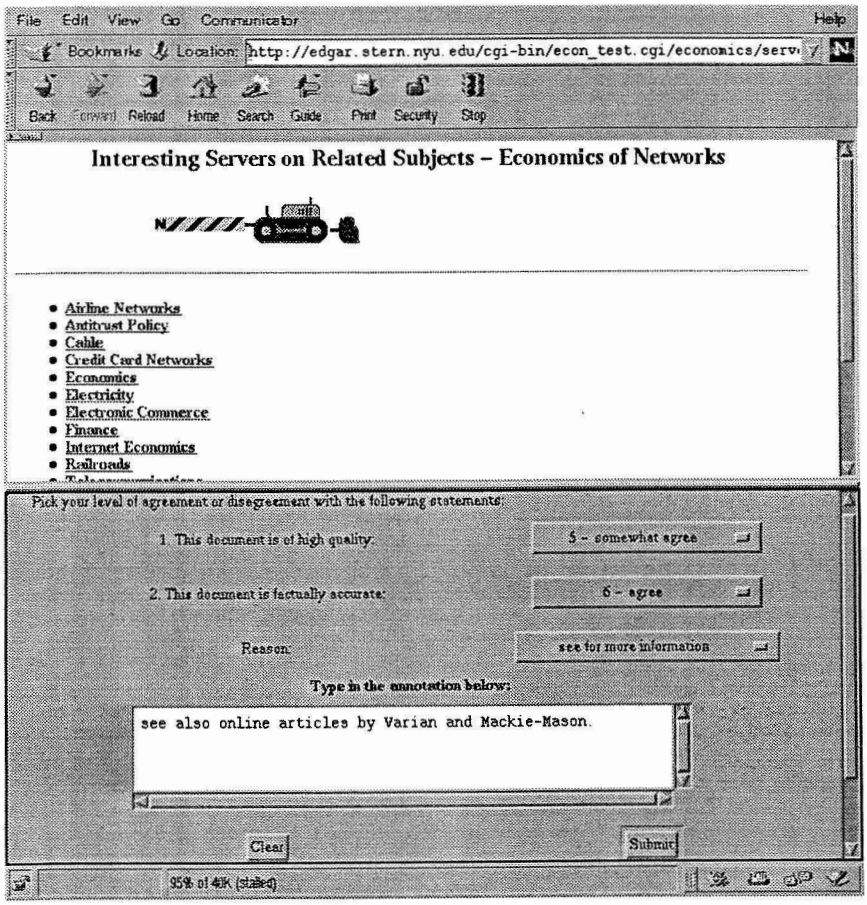

Figure 4: Document Layer: Creating a New Annotation

authenticated system will gain a sort of 'brand-name' recognition causing their notes to gain readership.

To aid in knowledge visualization, we use various small icons to denote appraisals and comments on the documents and convey them quickly to users. Figure 5 shows the complete range of possible icons that can be attached to the Retrieval Layer.

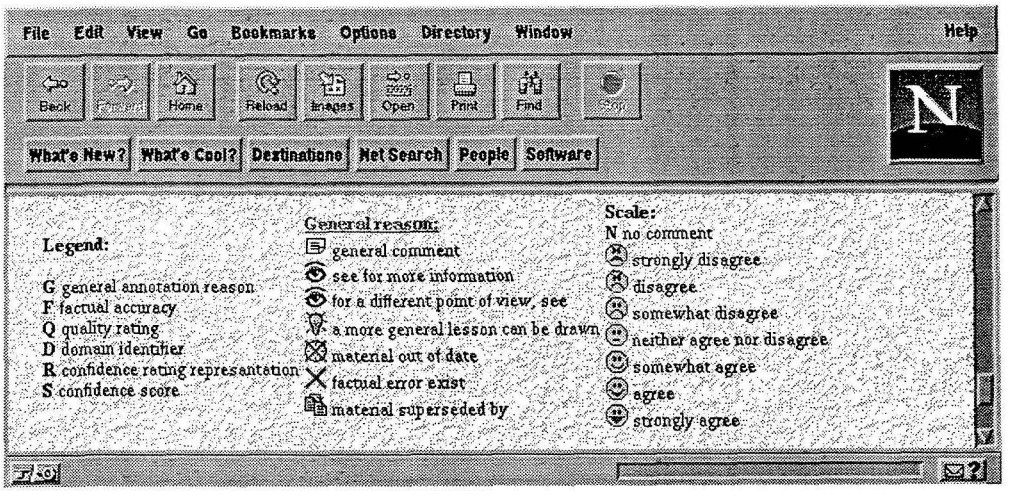

Figure 5: The Icon Legend

5 Annotate! System Architecture

Two key data stores, *discussion* data and *session* data, underly the Annotate! system. In this section we describe these data structures and show how they relate to the interface layers a user encounters in a search session.

5.1 The Discussion Data Store

The discussion data store stores the appraisal ratings, the free-text annotations of specific documents, and the reason for annotating (document out of date, a 'see' reference to another document, and so on). The discussion store is a hybrid of the originally published ('core') document with zero or more annotations.

Figure 6 presents the high-level view of the relationship between the discussion instances and the information retrieval interface layers (Query, Retrieval, and Document). The Document layer acts as a receptacle to collect user annotations. When an annotation event occurs, the discussion data store grows. This growth in turn may alter the look and feel of the Retrieval layer depending on simple trigger rules (refer to Section 4.1 for a full description).

Annotations add value to existing data: a legacy HTML or ASCII document is now coupled with annotations adding value to the existing document base. The annotations help users to refine their search by filtering on the annotation categories and enables collaborative (social) filtering as discussed theoretically by Avery and Zeckhauser (Avery and Zeckhauser, 1997). The annotations help users to socialize through allowing asynchronous collaboration, internalize or combine knowledge through looking at user defined annotations that guide to other sources, and support externalization by providing a mechanism for expressing annotations. Anonymous but authenticated annotations identifying

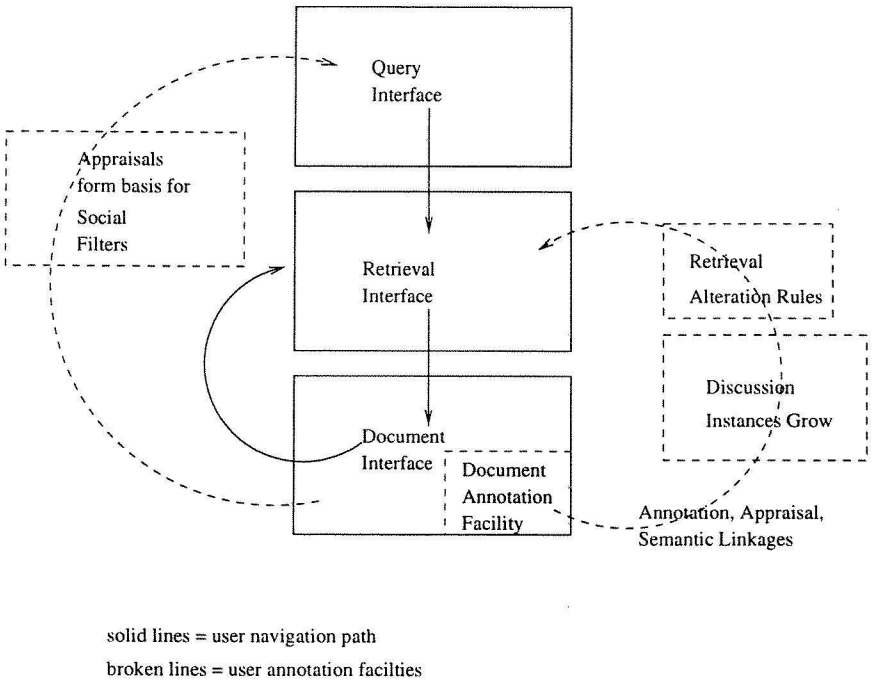

Figure 6: The Discussion Data Store influences, or is influenced by, all of the `Annotate!` Interface layers

the authors' workgroup can increase trust within an organizational setting.

Finally, and most importantly, the discussion instances *leverage the weaknesses of conventional full text information retrieval.* The `Annotate!` system anticipates that the results of full-text queries lack precision and are often spurious in the context of the original query. Annotations allow us to capture individuals' associative trails, and note interesting documents even if they did not match the original query; this has the potential to create new knowledge with subsequent system use. Since growth in the discussion data may create interface changes at every layer (Query, Retrieval and Document) there are many ways to alert users to new information relevant to their interests.

5.1.1 Policies to Manage the Discussion Instances

To realize the benefits of discussion data, organizations need to have supportive policies for knowledge management. Three policy decisions regard:

- Incentives and rewards for adding annotations and conversely, sanctions for non-participation. Without an explicit incentive scheme, Orlikowski (Orlikowski, 1992) demonstrated that Lotus Notes groupware was not well utilized at a management consulting company because its workers had little incentive to share information. The tradeoff to supplying annotation is the cost (time and effort) of constructing the notes versus the value of becoming an opinion leader and/or distinguishing oneself from

one's peer group.

- Level of anonymity of the annotator: anonymous, semi-anonymous (only group membership is identified, as in the `Annotate!` system), or non-anonymous. Prior research highlights the importance of anonymity but says little about group identification. For example, the social issues of anonymity and annotation have been explored in the Group Support System (GSS) setting by Connolly, Jessup and Valacich (Connolly et al., 1990). They show that anonymous readers are more likely to offer critical remarks.

- Controlling who may annotate. It is possible to limit the annotator population to designated experts in a given subject. For example, the `Annotate!` system can be extended to form a scholarly peer review system whereby domain experts annotate a draft manuscript.

5.2 The Session Data Store

The session data store keeps track of user queries, keywords, retrieval lists and the timings of the users' navigation through the document base. The relationship of the session instances to the search interface layers is shown in Figure 7.

As Figure 7 shows, the experimenter can use session data to evaluate `Annotate!` in a field setting. It is possible to write custom data analysis

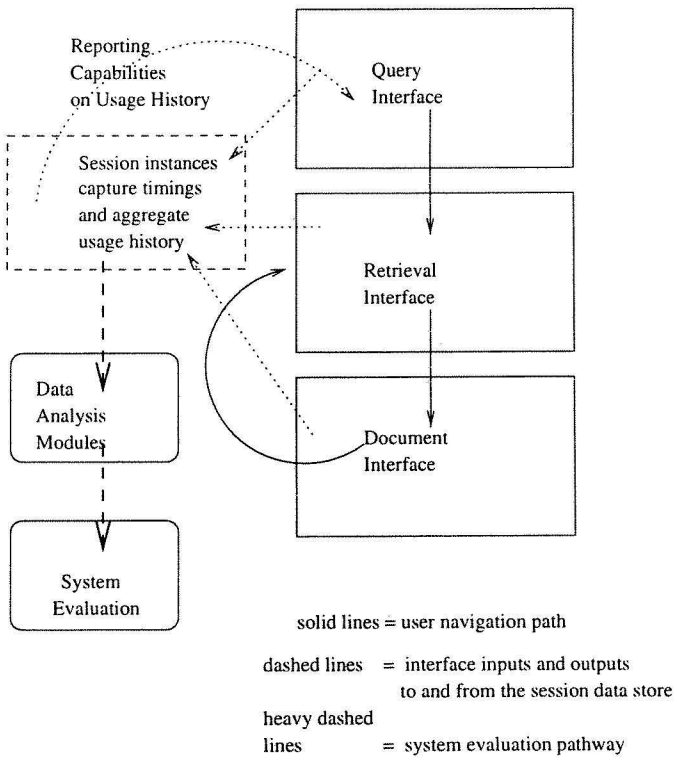

Figure 7: Session Instances Provide Interface Timings and Usage History

modules to perform more sophisticated tests of system usage; for example, in-depth analyses of document readership demographics and the times spent at the Query, Retrieval, and Document interface screens.

6 Evaluating the Annotate! System

We are presently evaluating Annotate! in an ongoing field trial at a federalist financial services firm. The control group makes use of the Excite full text search software and the experimental group uses the collaborative search features of Annotate! layered on top of Excite. We are collecting system variables, such as user navigation timings, document readership demographics, and annotation statistics and are developing analysis software as discussed in Section 5.2. We also collect qualitative data such as general user satisfaction measures and suitability of the system to the task at hand. The main focus of the research is to see if Annotate! increases document reach and range, and is judged to be more suitable to the task at hand. If these two important conditions hold, we can infer improved knowledge management: the information flow increases and the conversion of information to knowledge, judged subjectively by the recipient, is self-reported by the Annotate! system users.

7 Conclusions and Further Research

The WWW and Internet technologies enable new ways of implementing KMSS. `Annotate!` provides one mechanism to support knowledge management in federated organizations focusing on documents as repositories of relevant information for knowledge creation and use. Federated organizational forms are becoming more prevalent in a knowledge economy. The WWW and Intranet facilitate distributed document publishing necessitating effective storage,retrieval and KM mechanisms. KMSS should be designed to fit the organization form and enable organizations to implement policies for effective knowledge sharing.

Annotations improve the overall semantics of Web document (ASCII or HTML) by declaring user values and beliefs formally about documents. Annotate and Intranets increase *knowledge throughput* by increasing the flow of relevant information across business units. Even when users pursue documents irrelevant to the original query, the possibility of capturing subjective reactions will help in this regard. `Annotate!` begins to instantiate Nonaka's ideal of the *knowledge network* through provision of recommendations and navigation assistance. Furthermore by helping to increase the knowledge value of document repositories which span many business groups, `Annotate!` is designed to increase the interoperability of federated document collections which is a recent focus of research (Paepcke et al., 1998).

Ultimately such a system's effective use will be predicated on organizational policies and choices users make to define their own ontology. As we apply this tool in organizational settings, our current research examines incentives, authentication, anonymity and the impact of other policy choices on system use and effectiveness. Specifically we are modelling knowledge as a "collective organizational good", examining different levels or authentication, anonymity and policy choices on system use and effectiveness. Tools like `Annotate!` enable us to easily collect data and study the diffusion and sharing of know-how in organizations through electronic means.

References

Avery, C. and Zeckhauser, R. (1997). Recommender systems for evaluating computer messages. *Communications of the ACM*, 40(3):88–89.

Baldwin, C. Y. and Clark, K. B. (1997). Managing in an age of modularity. *Harvard Business Review*, pages 84–93.

Connolly, T., Jessup, L. M., and Valacich, J. S. (1990). Effects of anonymity and evaluative tone on idea generation in computer-mediated groups. *Management Science*, 36(6):689–703.

Davenport, T. H. and Prusak, L. (1998). *Working Knowledge: How Organizations Manage What They Know.* Harvard Business School Press, Boston, MA.

Ginsburg, M. and Duliba, K. (1997). Enterprise-level groupware choices: Evaluating Lotus Notes and intranet-based solutions. *CSCW: The Journal of Collaborative Computing*, 6:201–225.

Koda, T. and Maes, P. (1996). Agents with faces: The effects of personification of agents. In *Proceedings of HCI'96*, London.

LaLiberte, D. (1998). WWW collaboration projects. Technical report, UIUC, http://www.hypernews.org/HyperNews/get/www/annotations.html.

Nonaka, I. (1994). A dynamic theory of organizational knowledge creation. *Organization Science*, 5(1):14–37.

O'Leary, D. E. (1998). Enterprise knowledge management. *IEEE Computer*, 31(3):54–61.

Orlikowski, W. (1992). Learning from Notes: Organizational issues in groupware implementation. *Proceedings of CSCW 1992.*

Pejtersen, A. M. (1998). Semantic information retrieval. *Communications of the ACM*, 41(4):90–92.

Phelps, T. A. and Wilensky, R. (1996). Toward active, extensible, networked documents: Multivalent architecture and applications. In *Proceedings of the 1st ACM International Conference on Digital Libraries*, pages 100–108, Bethesda, MD. ACM.

Ross, J. W. and Rockart, J. F. (1996). Enabling new organizational forms: A changing perspective on infrastructure. *Proceedings of the International Conference on Information Systems (ICIS).*

Svenonius, E. (1986). Unanswered questions in the design of controlled vocabularies. *Journal of the American Society for Information Science*, 37(5):331–340.