Knowledge portals and the emerging digital knowledge workplace

by R. Mack
Y. Ravin
R. J. Byrd

A fundamental aspect of knowledge management is capturing knowledge and expertise created by knowledge workers as they go about their work and making it available to a larger community of colleagues. Technology can support these goals, and knowledge portals have emerged as a key tool for supporting knowledge work. Knowledge portals are single-point-access software systems intended to provide easy and timely access to information and to support communities of knowledge workers who share common goals. In this paper we discuss knowledge portal applications we have developed in collaboration with IBM Global Services, mainly for internal use by Global Services practitioners. We describe the role knowledge portals play in supporting knowledge work tasks and the component technologies embedded in portals, such as the gathering of distributed document information, indexing and text search, and categorization; and we discuss new functionality for future inclusion in knowledge portals. We share our experience deploying and maintaining portals. Finally, we describe how we view the future of knowledge portals in an expanding knowledge workplace that supports mobility, collaboration, and increasingly automated project workflow.

All human work, even the most physical labor, involves cognitive capabilities, but the hallmark of human work in the latter part of the twentieth century emphasizes *knowledge work*—solving problems and accomplishing goals by gathering, organizing, analyzing, creating, and synthesizing information and expertise. Knowledge work is performed by individuals who belong to communities of interest, where knowledge is shared and accumulated. *Knowledge management* (KM) refers to the methods and tools for capturing, storing, organizing, and making accessible knowledge and expertise within and across communities. Communities of interest may be scientific, academic, business-oriented, or government-based. We focus here on the corporate environment, since this is where KM is most self-consciously addressed, and where supporting technologies are expanding most rapidly.

At the broadest level (to paraphrase Prusak[1]), KM refers to all the tools, technologies, practices, and incentives deployed by an organization to "know what it knows" and to make this knowledge available to people who need to know it when they need to know it. At the individual or team level, the KM flow is a cycle in which solving a problem leads to new knowledge, initially tacit (that is, known but unexpressed), and then made explicit when experiences are documented, distributed, and shared (via databases, e-mail, or presentations). Once explicit, the knowledge is used by others for solving new problems.[2,3] The application of the explicit knowledge to a new problem creates new tacit knowledge, with the potential of initiating a new KM cycle. In this general cycle lie a host of technical, social, and human-computer interaction issues. In this paper we focus on the technology and, specifically, on what have come to be called knowledge portals.

0018-8670/01/$5.00 © 2001 IBM

From information portals to knowledge portals

The term "portal" is used quite ambiguously, especially because it evolved over time and became commonplace. Portals started as applications, typically Web-based, providing a single point of access to distributed on-line information, such as documents resulting from a search, news channels, and links to specialized Web sites. To facilitate access to large accumulations of information, portals quickly evolved to include advanced search capabilities and organizing schemes, such as taxonomies. Because of their emphasis on information, these first-generation portals are often called *information portals*. Information portals provide a valuable service on the Internet, by selecting, organizing, describing, and sometimes evaluating, useful sites. Yahoo![4] was one of the first and is still one of the most popular public-domain, Web-based portals. The recent proliferation of portals may seem to undermine the original intent of single access, but in fact, this circumstance emphasizes that portals are defined with respect to a community of users who share common tasks and interests. (Consider, for example, the viewpoint of a shopping consumer versus that of a professional engaged in researching a topic for a report.) This is especially true for internal corporate portals, where different functional and organizational groups and lines of business may have substantially different needs for information access and organization. Examples include sales and marketing, best practices, competitive intelligence, research and development, and general corporate resources. Specialized portals in the corporate sector are sometimes called *vortals*, for vertical portals, since they provide in-depth capabilities that are highly focused on a vertical segment of an organization or field.[5]

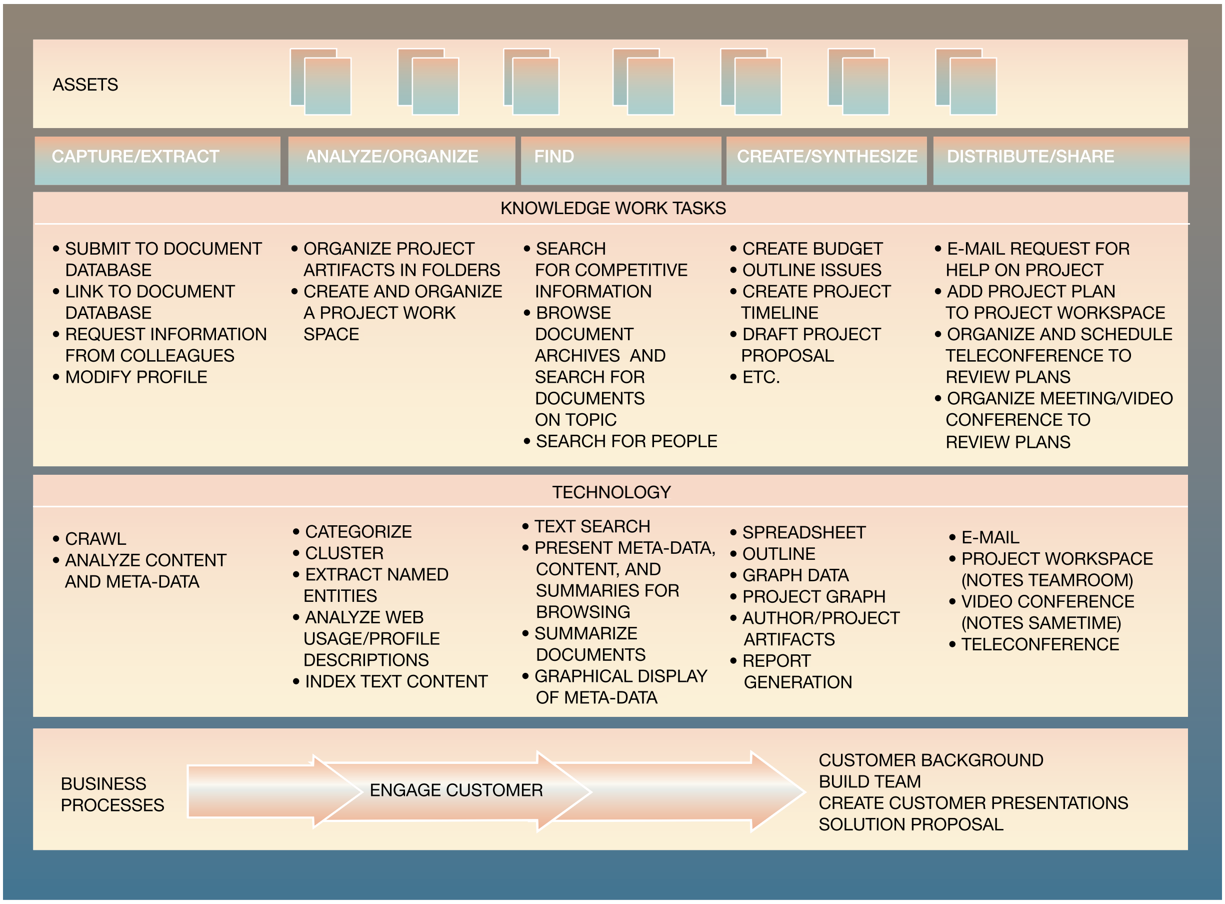
We refer to information portals used by knowledge workers as *knowledge portals* (or K Portals for short) to differentiate this KM role and usage from other portal roles, such as consumer shopping or business-to-business commerce. K Portals are rapidly evolving into broad-based platforms for supporting a wide range of knowledge worker (KW) tasks. We refer to the broad-based platforms as the *knowledge workplace* to draw attention to the importance of supporting the full range of knowledge work tasks within an integrated and unified context of use. In the next three sections of this paper, we focus on the information accessing and organizing role of portals and on how this role relates to the broader spectrum of knowledge work tasks. We describe component technologies and end-user functions, drawing heavily on our experience building a platform for portal systems. The section succeeding those ("Knowledge portals in an expanding knowledge workplace") discusses the evolution of K Portals and evokes themes that are covered in other papers in this issue.

Knowledge work and the role of portals

Portals serve tasks performed by knowledge workers, and we depict these tasks in the high-level view in Figure 1. Most broadly described, KWs gather information relevant to a task, organize it, search it, and analyze it, synthesize solutions with respect to specific task goals, and then share and distribute what has been learned with other KWs. The tasks are illustrated concretely in Table 1, which describes a "day in the life" of a consultant involved in the initial steps of engaging with a customer in a marketing or a consulting context. We use this scenario as the basis for portal technologies throughout this paper.

The consultant (and KW) goes through several task steps, starting with gathering information about the customer, the industry, or the business field relevant to the engagement, as well as about the products and the services available to meet the customer's needs. References might also be sought to colleagues who might have useful expertise to share. Using several tools, the KW searches internal and Web information resources for electronic and nonelectronic artifacts, often generated as a result of previous projects and distributed in a variety of ways. Searching is done both explicitly, using the portal search functions, and possibly implicitly, by creating or modifying a profile of current interests that is used to automatically find, and notify users of, potentially relevant information. An explicit search can involve formulating a query, reviewing search results, requesting "more documents like this," or browsing taxonomies that organize documents into topics. Over time, the KW acquires information relevant to the customer engagement and may create a dedicated project workplace in which to collect and organize these resources. This workplace supports further project activities, such as creating presentations or carrying out analyses needed as input for proposals, budgets, and project timelines. These activities involve soliciting information from other colleagues and experts via e-mail, scheduling meetings and teleconferences, distributing various artifacts, and saving information in a project workplace (e.g., Lotus Notes** TeamRoom) for review and coauthorship.

Figure 1 Knowledge work tasks, with examples of supporting technology



ASSETS

	CAPTURE/EXTRACT	ANALYZE/ORGANIZE	FIND	CREATE/SYNTHESIZE	DISTRIBUTE/SHARE
KNOWLEDGE WORK TASKS	• SUBMIT TO DOCUMENT DATABASE • LINK TO DOCUMENT DATABASE • REQUEST INFORMATION FROM COLLEAGUES • MODIFY PROFILE	• ORGANIZE PROJECT ARTIFACTS IN FOLDERS • CREATE AND ORGANIZE A PROJECT WORK SPACE	• SEARCH FOR COMPETITIVE INFORMATION • BROWSE DOCUMENT ARCHIVES AND SEARCH FOR DOCUMENTS ON TOPIC • SEARCH FOR PEOPLE	• CREATE BUDGET • OUTLINE ISSUES • CREATE PROJECT TIMELINE • DRAFT PROJECT PROPOSAL • ETC.	• E-MAIL REQUEST FOR HELP ON PROJECT • ADD PROJECT PLAN TO PROJECT WORKSPACE • ORGANIZE AND SCHEDULE TELECONFERENCE TO REVIEW PLANS • ORGANIZE MEETING/VIDEO CONFERENCE TO REVIEW PLANS
TECHNOLOGY	• CRAWL • ANALYZE CONTENT AND META-DATA	• CATEGORIZE • CLUSTER • EXTRACT NAMED ENTITIES • ANALYZE WEB USAGE/PROFILE DESCRIPTIONS • INDEX TEXT CONTENT	• TEXT SEARCH • PRESENT META-DATA, CONTENT, AND SUMMARIES FOR BROWSING • SUMMARIZE DOCUMENTS • GRAPHICAL DISPLAY OF META-DATA	• SPREADSHEET • OUTLINE • GRAPH DATA • PROJECT GRAPH • AUTHOR/PROJECT ARTIFACTS • REPORT GENERATION	• E-MAIL • PROJECT WORKSPACE (NOTES TEAMROOM) • VIDEO CONFERENCE (NOTES SAMETIME) • TELECONFERENCE

BUSINESS PROCESSES

ENGAGE CUSTOMER

CUSTOMER BACKGROUND
BUILD TEAM
CREATE CUSTOMER PRESENTATIONS
SOLUTION PROPOSAL

Figures 2 and 3 show screen shots of a K Portal built for internal use in IBM Global Services. This portal provides software support for several of the high-level KW tasks we have just described. The home page in Figure 2 is divided into several sections, some typical of Web pages (e.g., an identifying header or a footer with links to other pages on the Web site) and others specific to this portal. The left frame has a text entry field for conducting a search and filters for restricting search to a selected category shown in the center of the Web page (i.e., "Intellectual Capital and Assets," "Biographies," etc.). It also includes a document filter whose value represents the dominant type of document content (e.g., "projects" or "people"). In the central area of the home page are four high-level taxonomies that organize documents in ways relevant to KWs in IBM Global Services. Below the taxonomy area are bulletin board entries and top documents accessed by colleagues.

KWs can carry out free-text searches, navigate down one or more of the taxonomies, or combine a search with category and document-type restrictions. For example, Figure 3 shows a list of documents obtained by navigating to "Engagement Models," a subcategory branch of "Intellectual Capital and Assets," "Finance and Insurance," and "Financial Markets Solutions." (The subcategory path is shown at the top of the middle frame.) Documents such as those returned in Figure 3 can also result from inputting text terms expressing topics of interest into the search field in the upper left corner to initiate a search. A text search can be restricted to a specific category. This capability is important because many of the categories contain thousands of documents. Each document returned is summarized with a title, a link to the full document, an abstract, and indicators of document size and type. Abstract and size are useful for mobile users who may not want to download large

Table 1 Knowledge work scenario: Consultant involved in initial steps of a customer engagement

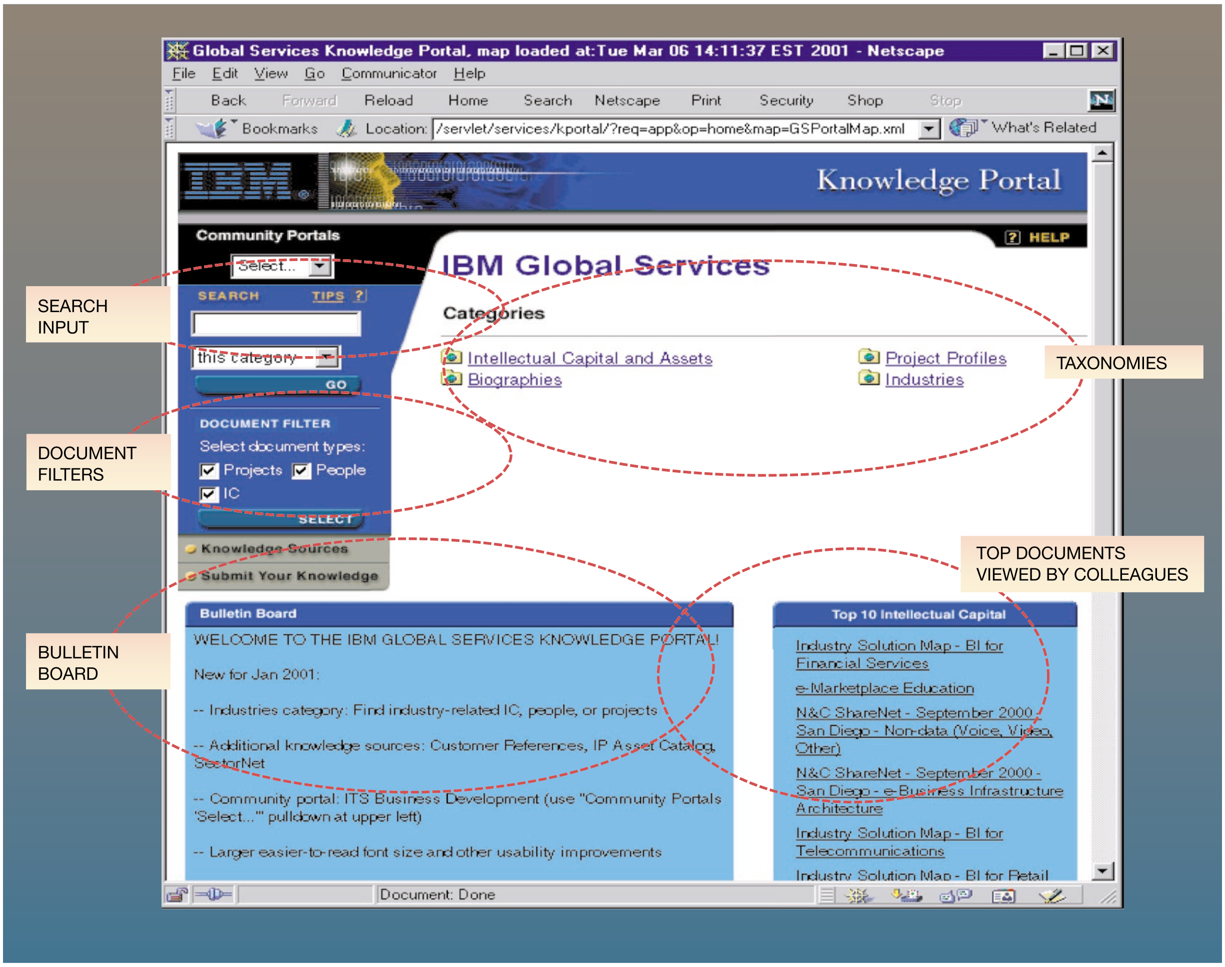
Steps	Actions
1	Customer representative **calls** knowledge worker (KW) named "Karen," a consulting services practitioner for "KnowledgeAdventures," informing her of a customer's interest in developing a document management system based on her company's products and services. Karen begins the first steps in a Customer Engagement. Karen and the customer representative **schedule a meeting** with customer CIO and technical staff to understand what they need.
2	Karen uses the portal to **find documents** relating to the customer, by looking into the "Engagement Life Cycle" taxonomy and **navigating down the category path** (Engagement Life Cycle → document management systems → customer references). She also finds a few digitized marketing **videos** on the company's product line, from a trade show. She downloads these documents to her workstation. Karen also **modifies her profile** to add the new customer name and descriptions of the problem to solve, and **requests notification** of information from external Internet sources on these topics.
3	Karen also needs to get help from experts in the company who know about the customer and document management systems. The portal returns **resumes** of other practitioners (in the Biography category) who cite document management as an expertise. Karen does not know most of these people (due to the high turnover in the services organization) and is unsure about their level of experience. She needs advice from colleagues.
4	Karen creates a **project workspace**, and **fills in a project template** with a set of categories representing phases of the project. This space will contain various project artifacts she anticipates gathering or creating, such as information about competitive products and technologies, skills and resources, existing assets based on prior engagements in the document management product domain, statements of work, etc. She **transfers documents** from her workstation to the workspace, to the appropriate categories (customer reference, competitive product information, etc.).
5	Karen sends off an **e-mail** note soliciting advice and interest from a set of colleagues that she either knows personally or has found via the resumes she gathered. She **schedules a teleconference**; her assistant **establishes a call-in number**, and **checks the schedules** of the people contacted. Karen **notices** that two of them are at a company site where she plans to be next week, and she wants to **find out** if they are available to meet in person.
6	In preparation for beginning the project, Karen **drafts a presentation** describing the customer's needs, the company's document management products and services, and **outlines a plan**. She puts the presentation in the project workspace, **alerts colleagues** that it exists, and schedules a conference to review it.

documents without knowing more about their content. An attachment indicator is useful too, since many Lotus Notes documents contain minimal text and serve as containers for attached documents.

Once a KW has gathered a set of documents relevant to a task, other tasks come into play, requiring support beyond searching and browsing, for authoring presentations and collaboration. Authoring and collaboration tools are not currently launchable from the K Portal described in Figures 2 and 3. Another IBM KM tool supporting both portal functions and certain types of collaboration is shown in Figures 4 and 5. The intellectual capital management (ICM) AssetWeb is a Notes-based application, originally developed for internal use within IBM Global Services.[6] It is now also available externally and has garnered acclaim as a KM tool in industry reviews.[7] The ICM AssetWeb uses Notes categories and TeamRooms (group document databases) to organize documents manually. Figure 5 shows an example of a TeamRoom. The left panel lists options for viewing documents in the repository by categories, chronology, or authorship, like any information portal, but because the ICM AssetWeb is built on Lotus Notes, it has access to the larger application context of Notes, with tools for collaboration and communication, including e-mail and calendars.

Portals support KWs as a community. Earlier prototype versions of the IBM Global Services K Portal shown in Figure 2, specialized for smaller "e-business" communities, listed references to news items, names of new hires, and icons pointing to feature stories, all of which were of potential interest to the practitioner community served by the portal, helping to build and support members of that community. The community is scattered across geographies, with many practitioners working from home offices or on the road. Featuring new employees electron-

Figure 2 Example knowledge portal home page, showing bulletin board, frequently accessed documents, and links to multiple taxonomies and text search input field


Global Services Knowledge Portal, map loaded at:Tue Mar 06 14:11:37 EST 2001 - Netscape
File Edit View Go Communicator Help
Back Forward Reload Home Search Netscape Print Security Shop Stop
Bookmarks Location: /servlet/services/kportal/?req=app&op=home&map=GSPortalMap.xml
What's Related
Knowledge Portal
Community Portals
Select...
HELP
IBM Global Services
SEARCH TIPS ?
this category
GO
Categories
Intellectual Capital and Assets
Biographies
Project Profiles
Industries
DOCUMENT FILTER
Select document types:
☑ Projects ☑ People
☑ IC
SELECT
Knowledge Sources
Submit Your Knowledge
Document: Done
SEARCH INPUT
DOCUMENT FILTERS
BULLETIN BOARD
TAXONOMIES
TOP DOCUMENTS VIEWED BY COLLEAGUES

Bulletin Board

WELCOME TO THE IBM GLOBAL SERVICES KNOWLEDGE PORTAL!

New for Jan 2001:

-- Industries category: Find industry-related IC, people, or projects

-- Additional knowledge sources: Customer References, IP Asset Catalog, SectorNet

-- Community portal: ITS Business Development (use "Community Portals 'Select...'" pulldown at upper left)

-- Larger easier-to-read font size and other usability improvements

Top 10 Intellectual Capital

Industry Solution Map - BI for Financial Services

e-Marketplace Education

N&C ShareNet - September 2000 - San Diego - Non-data (Voice, Video, Other)

N&C ShareNet - September 2000 - San Diego - e-Business Infrastructure Architecture

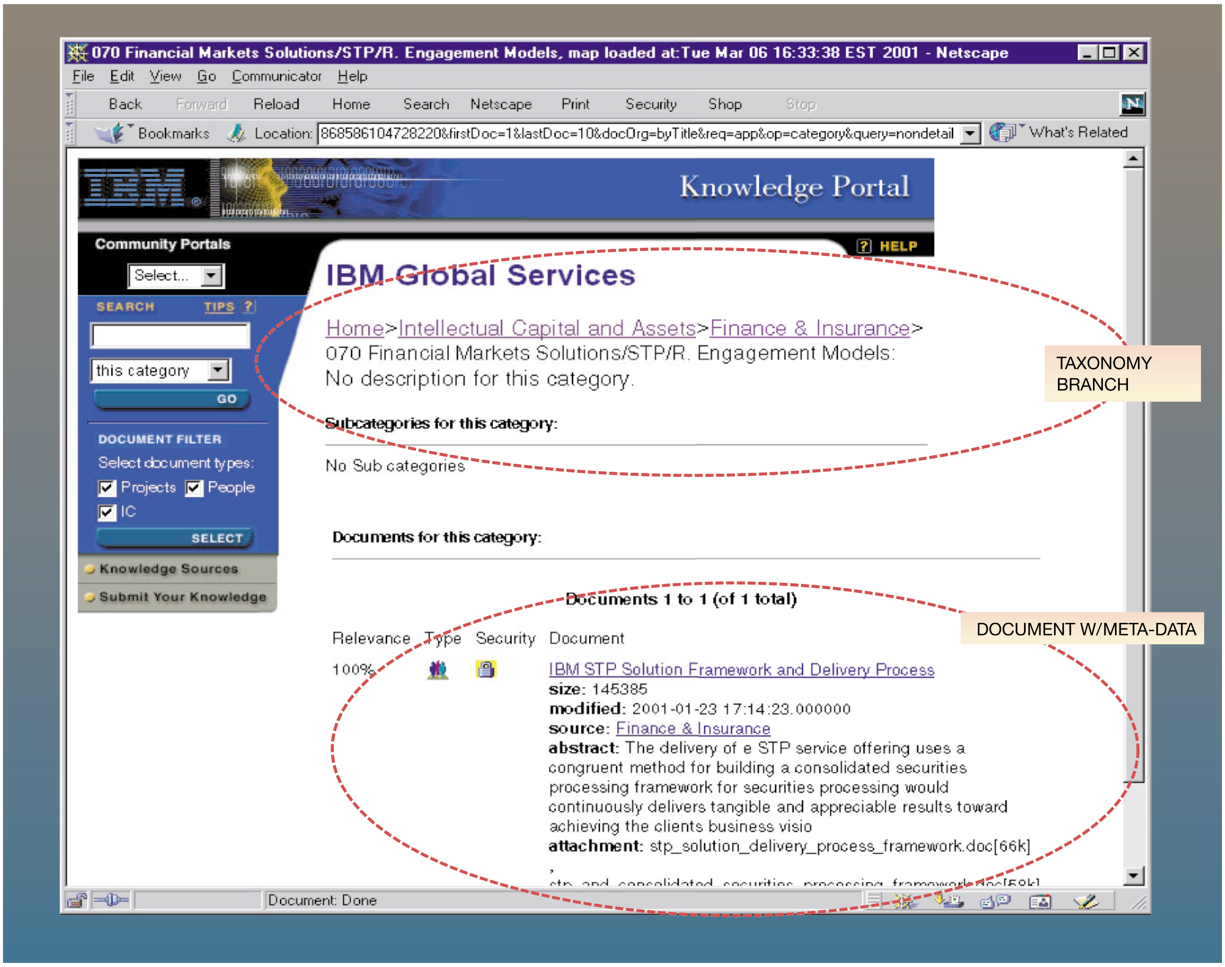
Industry Solution Map - BI for Telecommunications

Industry Solution Map - BI for Retail

ically fulfills an important social role, serving as a virtual welcome and introduction to the rest of their colleagues. Links to resources and biographical information help familiarize KWs with the community and are of special value to new hires. Bulletin boards, frequently accessed documents (shown in Figure 2), highlighted news, and success stories help shape the corporate culture and values, giving recognition and acknowledgment to successful employees, while creating models for others. These features are particularly important in a highly competitive, geographically dispersed profession with high turnover. Similarly, portals are known to be very important in merger and acquisition situations because they can bring together different corporate cultures to a single point of access.

To remain vital and current, both the IBM Global Services K Portal and the ICM AssetWeb require a variety of knowledge and content management processes (also discussed in the fifth section under "Portal management"). These processes include oversight of document gathering, indexing, and categorization. The reliance of a KW on the information available through the portal raises important concerns about the coverage and quality of the information sources. Higher-level KM processes include dedicated "core teams" that evaluate the quality of intellectual capital submitted to the portal. The document management process of the ICM AssetWeb includes review, classification, and certification of documents by dedicated teams of subject-matter experts from the appropriate IBM Global Service lines of business. Se-

Figure 3 Example knowledge portal showing a branch of a taxonomy, with categorized documents, showing document meta-data


070 Financial Markets Solutions/STP/R. Engagement Models, map loaded at:Tue Mar 06 16:33:38 EST 2001 - Netscape
File Edit View Go Communicator Help
Back Forward Reload Home Search Netscape Print Security Shop Stop
Bookmarks Location: 868586104728220&firstDoc=1&lastDoc=10&docOrg=byTitle&req=app&op=category&query=nondetail
What's Related
Knowledge Portal
Community Portals
Select...
? HELP
SEARCH TIPS ?
this category
GO
DOCUMENT FILTER
Select document types:
☑ Projects ☑ People
☑ IC
SELECT
Knowledge Sources
Submit Your Knowledge
TAXONOMY BRANCH
DOCUMENT W/META-DATA
Document: Done

IBM Global Services

Home>Intellectual Capital and Assets>Finance & Insurance>
070 Financial Markets Solutions/STP/R. Engagement Models:
No description for this category.

Subcategories for this category:

No Sub categories

Documents for this category:

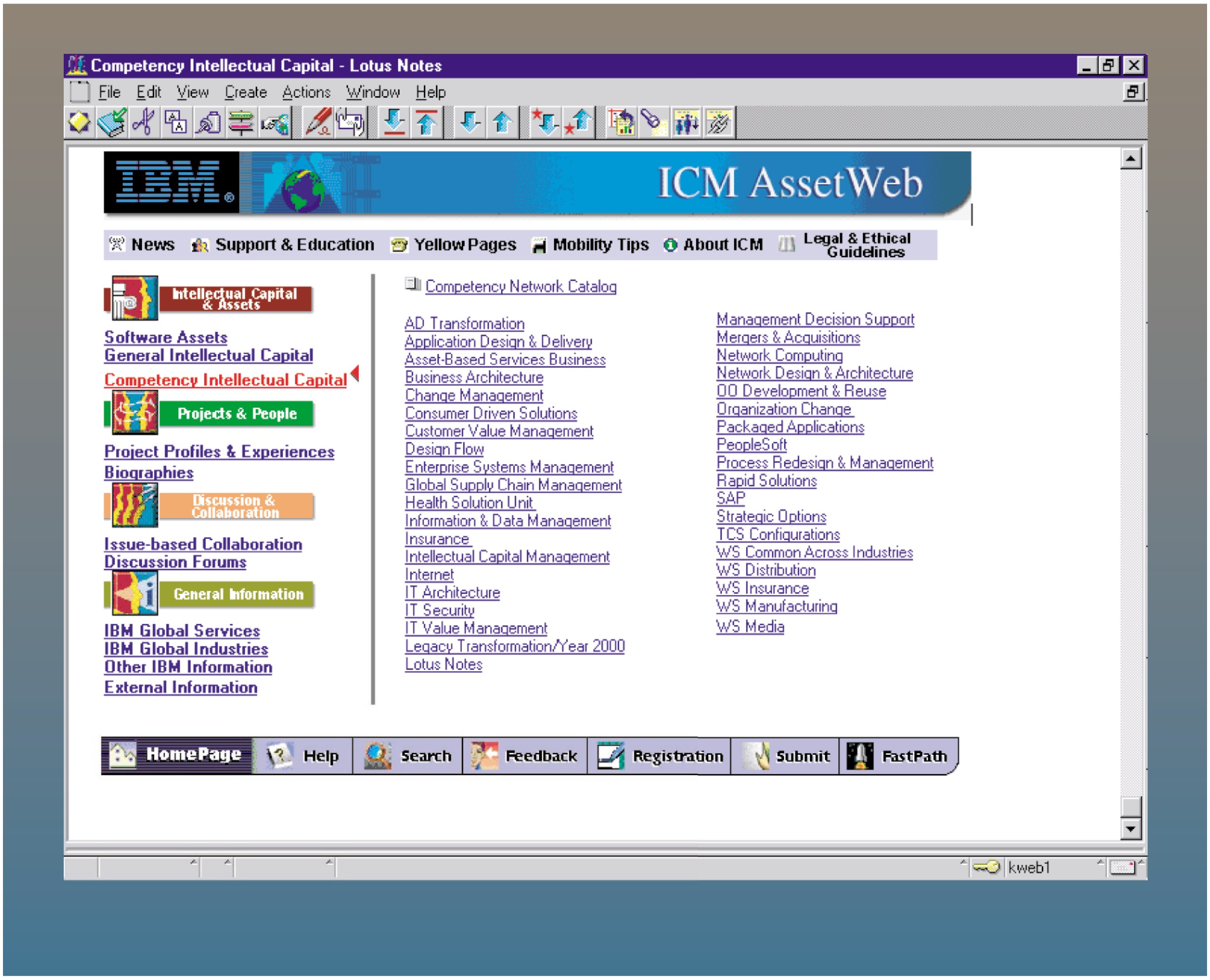
Documents 1 to 1 (of 1 total)

Relevance	Type	Security	Document
100%			IBM STP Solution Framework and Delivery Process **size**: 145385 **modified**: 2001-01-23 17:14:23.000000 **source**: Finance & Insurance **abstract**: The delivery of e STP service offering uses a congruent method for building a consolidated securities processing framework for securities processing would continuously delivers tangible and appreciable results toward achieving the clients business visio **attachment**: stp_solution_delivery_process_framework.doc[66k] , stp_and_consolidated_securities_processing_framework.doc[58k]

curity issues involved in accessing documents are also of concern. Access to documents is controlled by the document repositories themselves. Users need to enter a user ID (identifier) and password to see certain documents.

The IBM Global Services K Portal and the ICM AssetWeb systems complement each other. The portal is totally Web-based, lightweight, and focused on search and categorization. The middleware supporting it allows easy integration of new exploratory functions. The ICM AssetWeb, in contrast, includes collaboration and communication tools, some level of workflow to manage the content, and application development tools. The new generation of IBM KM product platforms, including the Lotus Discovery Server**[8,9] and WebSphere* Portal Server, integrate an expanded search function, including finding expertise, i.e., knowledgeable colleagues and potential team members, taxonomy generation tools, and more easily customized collaboration spaces (Lotus QuickPlaces**). Other vendor offerings, such as those from Plumtree Software[10] or Autonomy[11] offer similar capabilities. But we contend that there is room for more research and development to improve the quality of specific features, such as search, categorization, and support for collaboration, as well as for the effective integration of these features. Achieving these goals will lead to a much richer and more supportive knowledge workplace. We return to this point after we review in more depth the component technologies that we have outlined.

Figure 4 ICM AssetWeb showing a taxonomy branch of IBM Global Services “Knowledge Network” documents

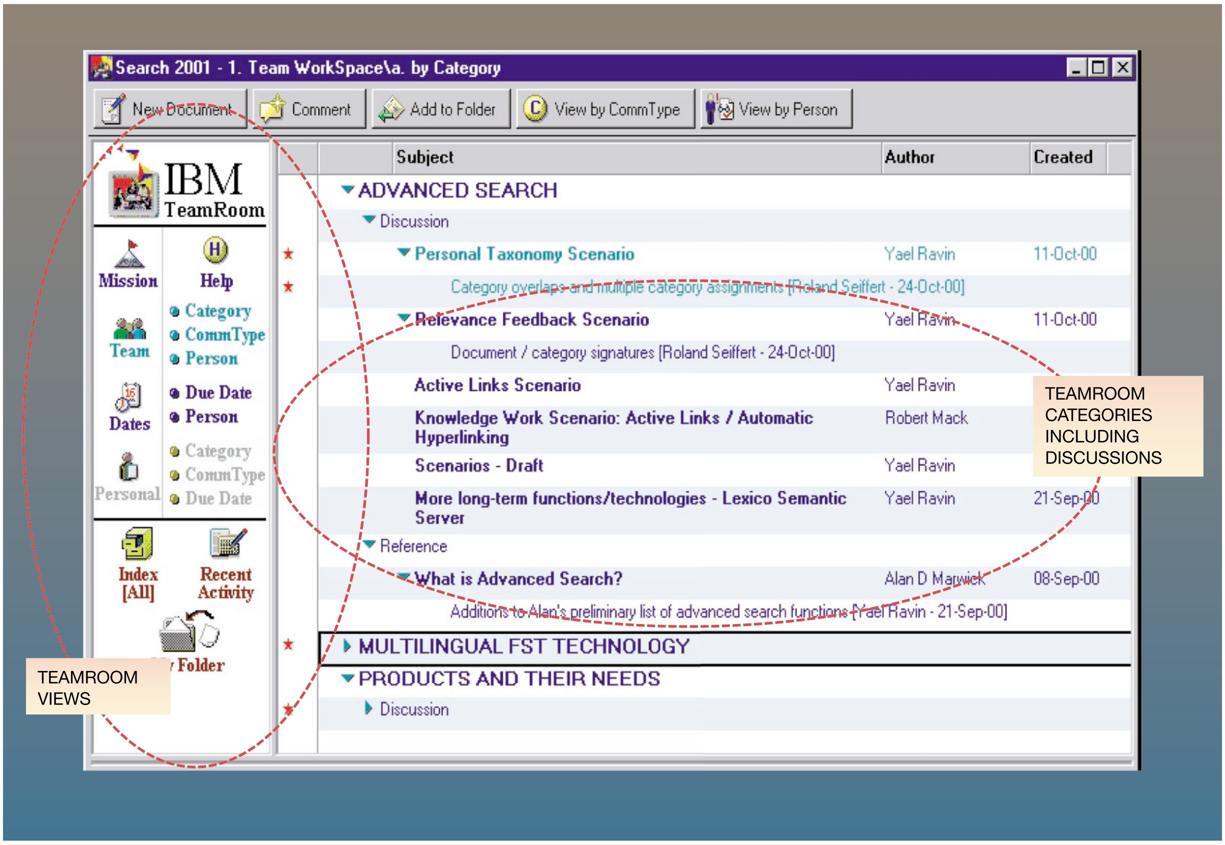

Competency Intellectual Capital - Lotus Notes
File Edit View Create Actions Window Help
ICM AssetWeb
News
Support & Education
Yellow Pages
Mobility Tips
About ICM
Legal & Ethical Guidelines
Intellectual Capital & Assets
Software Assets
General Intellectual Capital
Competency Intellectual Capital
Projects & People
Project Profiles & Experiences
Biographies
Discussion & Collaboration
Issue-based Collaboration
Discussion Forums
General Information
IBM Global Services
IBM Global Industries
Other IBM Information
External Information
Competency Network Catalog
AD Transformation
Application Design & Delivery
Asset-Based Services Business
Business Architecture
Change Management
Consumer Driven Solutions
Customer Value Management
Design Flow
Enterprise Systems Management
Global Supply Chain Management
Health Solution Unit
Information & Data Management
Insurance
Intellectual Capital Management
Internet
IT Architecture
IT Security
IT Value Management
Legacy Transformation/Year 2000
Lotus Notes
Management Decision Support
Mergers & Acquisitions
Network Computing
Network Design & Architecture
OO Development & Reuse
Organization Change
Packaged Applications
PeopleSoft
Process Redesign & Management
Rapid Solutions
SAP
Strategic Options
TCS Configurations
WS Common Across Industries
WS Distribution
WS Insurance
WS Manufacturing
WS Media
HomePage
Help
Search
Feedback
Registration
Submit
FastPath
kweb1

Knowledge portal technologies

Here we delve more deeply into the technologies integrated in portals, in some cases discussing issues and capabilities that exist only in research prototypes or in competitive products. We discuss topics roughly in the order of the high-level tasks schematized in Figure 1.

Capture and gather. Documents created in the course of performing knowledge work are typically stored in multiple places—file systems on individual workstations, Web sites on network servers, and document management systems such as Lotus Notes. In order to make content accessible to the portal base technologies and ultimately to users, documents need to be automatically gathered by the system, registered, managed, and analyzed. Documents are extracted via a process called *crawling*, which starts from a given URL (uniform resource locator) or another specific address, and then automatically and recursively follows all the links in each document. Content analyzers extract text and meta-data from each document as it is “crawled” and handle the particulars of different document formats. The IBM Global Services K Portal uses a specific technology called Grand Central Station (GCS), originally developed at the IBM Almaden Research Center[12] to crawl documents in Lotus Notes databases, and Web sites. In both cases, GCS extracts text and meta-data from documents in multiple formats, such as Lotus word pro-

Figure 5 Notes TeamRoom with document and discussion categories



Subject	Author	Created
ADVANCED SEARCH		
Discussion		
Personal Taxonomy Scenario	Yael Ravin	11-Oct-00
Category overlaps and multiple category assignments [Roland Seiffert - 24-Oct-00]		
Relevance Feedback Scenario	Yael Ravin	11-Oct-00
Document / category signatures [Roland Seiffert - 24-Oct-00]		
Active Links Scenario	Yael Ravin	
Knowledge Work Scenario: Active Links / Automatic Hyperlinking	Robert Mack	
Scenarios - Draft	Yael Ravin	
More long-term functions/technologies - Lexico Semantic Server	Yael Ravin	21-Sep-00
Reference		
What is Advanced Search?	Alan D Marwick	08-Sep-00
Additions to Alan's preliminary list of advanced search functions [Yael Ravin - 21-Sep-00]		
MULTILINGUAL FST TECHNOLOGY		
PRODUCTS AND THEIR NEEDS		
Discussion		

TEAMROOM CATEGORIES INCLUDING DISCUSSIONS

TEAMROOM VIEWS

cessing and business graphics applications, and the corresponding Microsoft office applications. For Lotus Notes documents, information is also extracted from attached documents. Extracted text and metadata are encoded in a standard XML (Extensible Markup Language) format across document types and made available for subsequent indexing and analysis processes.

There are at least two reasons for aggregating electronic information using a crawler. First, aggregating data makes it easier to create a centralized search index for a collection, enabling a search over all documents using a common search approach. Second, many useful methods for analyzing documents require analyzing the properties of document aggregates, as we discuss in the next subsection.

However, it is not always possible to carry out full-scale, automatic crawling. For example, a repository of documents, such as Dow Jones Interactive**, may be stored in a proprietary database system with an interface that controls access, preventing systematic crawling of its contents. It may not be possible for an external portal to access the information in the repositories systematically, as required for creating a search index within the portal. In this case, an alternative federated search strategy may be needed to create unified access to information across multiple repositories. In a federated search, a query specification created by a user is sent to multiple search engines, and the results are aggregated. Distributing the search and combining results in this way is technically challenging for several reasons (see Reference 13). A related situation arises where a single central index may be too large. In this case, the central index can be structured in multiple indices to allow more efficient parallel processing of smaller groups of document statistics. Once again, technology exists for distributing a query to all the indices in parallel, and then collecting the results and merging them.[14]

Access control to information is an important issue. Some of the information may have to be restricted to specific communities. Portals could accommodate this situation by simply not including restricted information in the search index or in categorized subcollections. However, this limitation would undermine the rationale of portals, which is to inform users of what information is available. One way to handle access restrictions is to provide summary information of sensitive documents but control access to the full content. In the IBM Global Services K Portal, search results return document titles and abstracts, including a link to the document in the repository where it is stored, with access subject to the access protocol of the repository, which may require users to log in to the repository with a password. An icon next to the document title in a search hit list indicates whether access is restricted and saves the user the annoyance of trying to access the document when it is not available. In some circumstances, even a document title may be too sensitive. Human resource documents may contain titles or abstracts that identify people and personal issues that would violate business policies and possibly privacy laws. In these cases, it is necessary to make clear access policies. It may be possible to create sanitized summaries of sensitive documents, sufficient to alert users to the existence of this information, while still protecting it.

Document analysis—Text analysis and feature extraction. Once the documents have been gathered, they must be analyzed so that their content is available for subsequent organization, retrieval, and use by the system and by KWs. In subsequent subsections, we present text analysis operations performed by the system, involving various forms of clustering, categorization, searching, navigation, and visualization of documents. Here we discuss the document analysis required in preparation for these operations.

As documents enter the portal system, they are stored for later retrieval and display. However, it is not useful to simply put the documents away in their raw form. Systems typically analyze the document content and store the results of that analysis so that subsequent use of the documents by the system and users will be more effective and efficient.

In order to operate on documents, we extract document features that give an indication of what documents are "about." Since documents contain text, the portal applies text analysis in order to extract *textual features*, which characterize the documents. At the lowest level, these features are characters and words. However, when it is important to manage the conceptual content of documents, we need to identify the entities referred to in the text—the things, people, places, organizations, dates, prices, etc.—that are specific to the domain from which the documents are drawn and that will make useful features for subsequent organization, search, and browsing operations. Certain operations will also require features consisting of relationships among these entities.

In addition to the textual features, which are intrinsic to the document (i.e., drawn from within it), there are also extrinsic features, whose source is outside the document. These features, also called *meta-data features*, include information about creation date, author, category assignment within a classification scheme, confidentiality, etc. Often, this meta-data information is gathered by the crawling process, and the crawled content is represented in XML format, with the meta-data features encoded by XML tags within the XML files. For some operations, the distinction between intrinsic and extrinsic features is irrelevant. Hence, in what follows, we will often use the word "feature" to refer to both textual features and meta-data features.

Since document text is a form of human language, a wide variety of linguistic analysis techniques can be used to find vocabulary and other language expressions that refer to domain entities and their relations. These expressions and the concepts they refer to reflect the conceptual content of the document collection. These expressions provide the features used for organizing and finding documents in portal systems. The simplest and most widespread type of feature used in current systems is simply the words in the text. These words are easy to obtain with simple tokenization technology. With the addition of straightforward processing such as morphological processing (e.g., combining plural nouns with singular ones) and stop-word processing (i.e., ignoring common words), word-based systems perform well in some operations, such as a document search.

However, in knowledge-based applications, such as taxonomy generation and navigation, it is important to identify features that reflect the domain-specific conceptual content of documents more accurately than simple words can. We have built a system, called Textract,[15–18] that can automatically process a document collection and identify expressions that are proper names (of people, places, organizations, etc.), domain terms, abbreviations, and various types of expressions such as dates and amounts of money. Further, domain experts can customize Textract so that it will also recognize various types of domain-specific entity references, such as telephone numbers and document IDs. The techniques that Textract uses depend on the analysis of patterns of symbols in text. This analysis capitalizes on the conventions and redundancy that are characteristic of the use of human language in documents. Thus, for example, over a large document collection, Textract is able to determine that the expressions "Senator Clinton," "Hillary Rodham Clinton," "Hillary," and even "the senator" are all references to the same person. Such information enables text analysis systems to better determine the topics of documents and to gauge the importance of entities that are referred to across the collection.

Beyond entity references, document analysis should also identify relationships among the entities. Textract uses the contexts in which expressions occur to find both statistical and lexical relations between the domain entities. The lexical relations (such as: ⟨Hillary Rodham Clinton : senator : New York State⟩) are found by doing a deeper linguistic analysis of the phrases and clauses in the text of the documents. Note that both the relations and the names of the relationships that link entities are discovered during document analysis. Statistical relationships among entities are found using various measures of the frequency with which they occur.[17]

Having entity references and relationships as textual features for characterizing the content of document collections is tremendously advantageous during the construction of knowledge portals by enabling operations that portals must perform. The following subsection discusses organization operations (clustering and categorization). Later subsections discuss search, query refinement, relevance feedback, and lexical navigation. Other operations such as summarization, glossary extraction, and question answering also depend crucially on the conceptual content of documents that these features reflect.

Document organization: Clustering and categorization. When the crawler has finished its gathering task, most often the result is an undifferentiated set of documents. As the number of documents under management grows, it becomes increasingly important to gather similar documents into smaller groups and to name the groups. This operation is *clustering*. All automatic clustering methods use features to determine when two documents are similar enough to be put into the same cluster. A typical approach taken is to represent a document as a vector of the features it contains and to compare the vectors for different documents. Variants of this approach optimize performance by ignoring features that occur too seldom, too often, or with distributions that do not allow them to effectively distinguish one document from another. For example, the feature for "IBM" would not be useful for clustering documents in an IBM internal portal.

It is almost impossible for the portal administrator (and domain expert) to know ahead of time how many clusters or which clusters are implied by the available documents. Nevertheless, there needs to be some way to control the operation of the clustering engine. Perhaps the most important control point is the choice of which documents are presented to the clusterer. For example, an administrator might choose to include formal documents such as reports or press releases, while excluding informal documents such as e-mail messages or chat room transcripts. The rationale for such decisions might be that the formal documents contain a more reliable account of the conceptual content of the domain, whereas the informal documents can be added to the resulting clusters later using a different technique, such as categorization. Depending on the system, clusterers can also accept parameters to control the sizes of clusters, the sensitivity of the similarity metric, or the total number of clusters. An important additional control point is the selection of features and their weights. Recall that the set of features available includes meta-data features such as document date, author, and assigned keywords. These can also affect the resulting set of clusters. In fact, one powerful use of extrinsic features might be to allow the clusterer to preserve some aspects of a previously existing category system by including category information among the features of the documents.

Rather than a flat space of clusters, some clustering engines are capable of building hierarchical structures containing clusters and subclusters. One approach taken is to accumulate similar documents into a cluster until some critical size is reached and to then split the cluster into two or more subclusters. Control points for such clustering engines include the critical size, the intracluster similarity metric, and the number of subclusters to build.

Once the clusterer has finished its work, the clusters must be named. *Cluster labeling* is the operation of inspecting the final cluster contents and choosing the best features to serve as names. The features used as labels are not necessarily the same as those used in the similarity metric. The requirement for labels is that they be easily understood by human users of the portal, evocatively characterize the documents in a cluster, and clarify the distinctions among neighboring clusters in a hierarchy.

An adequately labeled set of hierarchically organized clusters for a document collection is usually called a *taxonomy*, and the labeled clusters in the taxonomy are called *nodes*. It is a tall order for a clustering engine and labeler to get everything right totally automatically. As a consequence, systems that attempt to do automatic taxonomy generation usually incorporate a taxonomy editor so that the portal administrator or some other domain expert may craft a high-quality taxonomy based on the work of the automatic system components. Operations supported by a taxonomy editor include moving documents from one cluster to another, splitting or combining clusters, and manually assigning labels to clusters. The Lotus Discovery Server[9] provides a taxonomy generation tool based on the IBM Almaden Research Center's SABIO clustering technology.[19,20] An additional useful feature within taxonomy editing tools is document summarization. As the domain expert inspects document assignments to clusters and moves documents from cluster to cluster, it must be easy to discern the conceptual content of groups of documents without needing to read them in their entirety. Summarizers such as those described in the later subsection "Find" can produce sentential, keyword, or topic summaries that are suitable for this task.

Because document collections are not static, portals must provide some form of taxonomy maintenance.

Because document collections are not static, portals must provide some form of taxonomy maintenance. As new documents are added, they must be added to the taxonomy at appropriate places, using the classification technology described below. As the clusters grow, and especially as the conceptual content of the new documents changes over time, it may become necessary to subdivide clusters or to move documents from one cluster to another. Although less common, document deletions may also occur. For these reasons, it becomes appropriate to periodically reassess the taxonomy. As with taxonomy generation, this reassessment may be accomplished using both automatic and manual procedures. The automatic part, perhaps based on the same technology that spawned subclusters during taxonomy generation, can suggest when and how a cluster that has grown too large must be split. A portal administrator, using the taxonomy editor, can monitor and implement these suggestions and, in general, can periodically assess the health and appropriateness of the current taxonomy and document assignments within it.

As exemplified in the "Intellectual Capital/Finance and Insurance/ . . . Engagement Models/" taxonomy branch in Figure 3, a document classification scheme provides a powerful way for portal users to navigate through the document collection in their search for documents relevant to their information needs. Whether a classification scheme is based on an automatically generated taxonomy (e.g., one derived from the documents in the portal) or on an externally imposed taxonomy (e.g., one imposed by corporate management), it is crucial to be able to accurately assign documents to the taxonomy nodes. Such accuracy is important so that when users navigate to a node and access documents through it, they can expect that all the documents found are appropriate to the node and belong together. Clearly, in the case of automatic taxonomy generation, the clustering technology should meet this expectation, at least for the initial set of documents. However, for documents added to the portal after taxonomy generation—and for all documents in a portal with an externally imposed taxonomy—another mechanism is needed. Document categorization technology provides that mechanism.

The job of a document categorization system is to assign documents to categories, which are equivalent to the nodes in a taxonomy. In its simplest terms, a document categorization system operates in two steps. In the first step, the training step, the system inspects a set of previously categorized documents (the training set) and extracts a characterization of the documents in each category. This characterization, invariably based on the features found in the documents, is formatted and stored in a model. In the second step, the categorization step, the system processes one uncategorized document at a time. It extracts features from the document and compares them to the features stored for each category in the model. (Various optimization schemes can make these comparisons efficient to perform.) The result is a list of one or more categories to which the system thinks the new document should be assigned.

Extensive descriptions of a wide variety of approaches to categorization can be found in Baeza-Yates and Ribeiro-Neto.[21] The major differences among categorization systems concern the types of features they use, the way in which they represent the features associated with categories, and the way in which they compare document features with category features. For example, in the IBM Text Analyzer system, the features are words; they are associated with a category by means of "if-then" rules corresponding to a decision tree. Document features are compared to category features by means of a decision tree processor. In contrast, the IBM Global Services K Portal uses a K nearest neighbor approach, in which the comparison between document and category is done with a standard search engine. The categorization procedure uses features from the uncategorized document as a query against the set of training documents. The result of the search is a hit list of training documents. The category chosen for the uncategorized documents is the one associated with the majority of the highly ranked training documents on the hit list. The categorization system in IBM's original Intelligent Miner* for Text product[15] uses a centroid approach, in which the features are vocabulary items produced by Textract; the categories are represented by vectors consisting of their most salient features (one vector per category). This representation is similar to the feature vectors described above for document clustering engines. In the centroid approach, the comparison is essentially a vector-space comparison between a document feature vector and the category vectors.

These clustering and classification methods differ in their underlying algorithms, in how the tools associated with them are used, and in their effectiveness for given document domains. When discussing taxonomy generation, we pointed out the need for a taxonomy editor with which domain experts can review and repair decisions made by the automatic clustering and labeling machinery. These tools may require users to find training documents or define if-then rules, or do some combination of these two tasks. Similarly, categorization engines are not perfect, and some are more effective for some types of documents than others, e.g., Web documents versus documents produced by office productivity tools, versus news articles, which tend to be relatively unstructured. Fortunately, most categorization systems produce a rank associated with their category suggestions for a document. These ranks represent the degree of match between the features of the document and those of the categories of the model, and they correlate with the degree of confidence a user should have in the assignment of the document to the category.

To conclude, clustering and classification are very important organizing tools for portals, but it is clear that no one technique is best and that all techniques need domain expertise and some degree of administrative skill.

Find: Basic and advanced search. Once information is gathered and categorized, users can search it to find what they need using various techniques, from a basic text search to document result browsing interfaces on the Web, to more sophisticated search and browsing tools that we describe below.

The basic technique for retrieving documents by means of a query became widespread starting in the 1980s.[22] The process begins before search, when documents are scanned to produce an inverted index, a kind of dictionary that lists all the words appearing in the documents together with their locations. The index is the repository searched when a query is processed. Early systems tended to index only keywords, selected from the title or other meaningful fields in the documents. However, in the last 20 years, with more memory and cheaper storage, systems typically have full-text indexing of all the words and all occurrences. A query formulated by the user is usually lightly processed (e.g., stop words are removed) and sent to the search engine to be matched against the index. Many search algorithms are used for this matching. Most typically, the query is analyzed into a list of query terms, and the index is searched for documents that contain the query terms. The underlying assumption is that the user is interested in documents that contain the query terms and, more specifically, that documents containing frequent mentions of the query terms are more relevant. Several ranking algorithms for computing and sorting relevant documents have been developed. Many are based on a tf/df (term frequency divided by document frequency) formula, standing for the ratio between the frequency of a term in the document and the number of documents in the repository in which the term appears. This means that the contribution of a term to the document relevance is higher the more times the term is mentioned in the document, but this contribution is reduced if the term occurs in many other documents as well. Many systems refine this basic formula by normalizing for the length of the document, taking the order and proximity of the terms into account, or allowing for minor term variations (such as plural and singular forms of nouns), among other strategies.

Basic searching, as described here, is the most common method for finding information on line, yet often users do not find the information they are looking for. There are a number of reasons for this. The ever-increasing size of document collections increases the pool of potentially relevant documents. If the collection is heterogeneous, query words may be ambiguous—the same words may refer to different concepts in different domains. Finally, if the query is short (the average Web query is under three words long), it contains fewer terms and thus matches more documents. To compensate for these factors, we are developing advanced search techniques called prompted query refinement and relevance feedback.

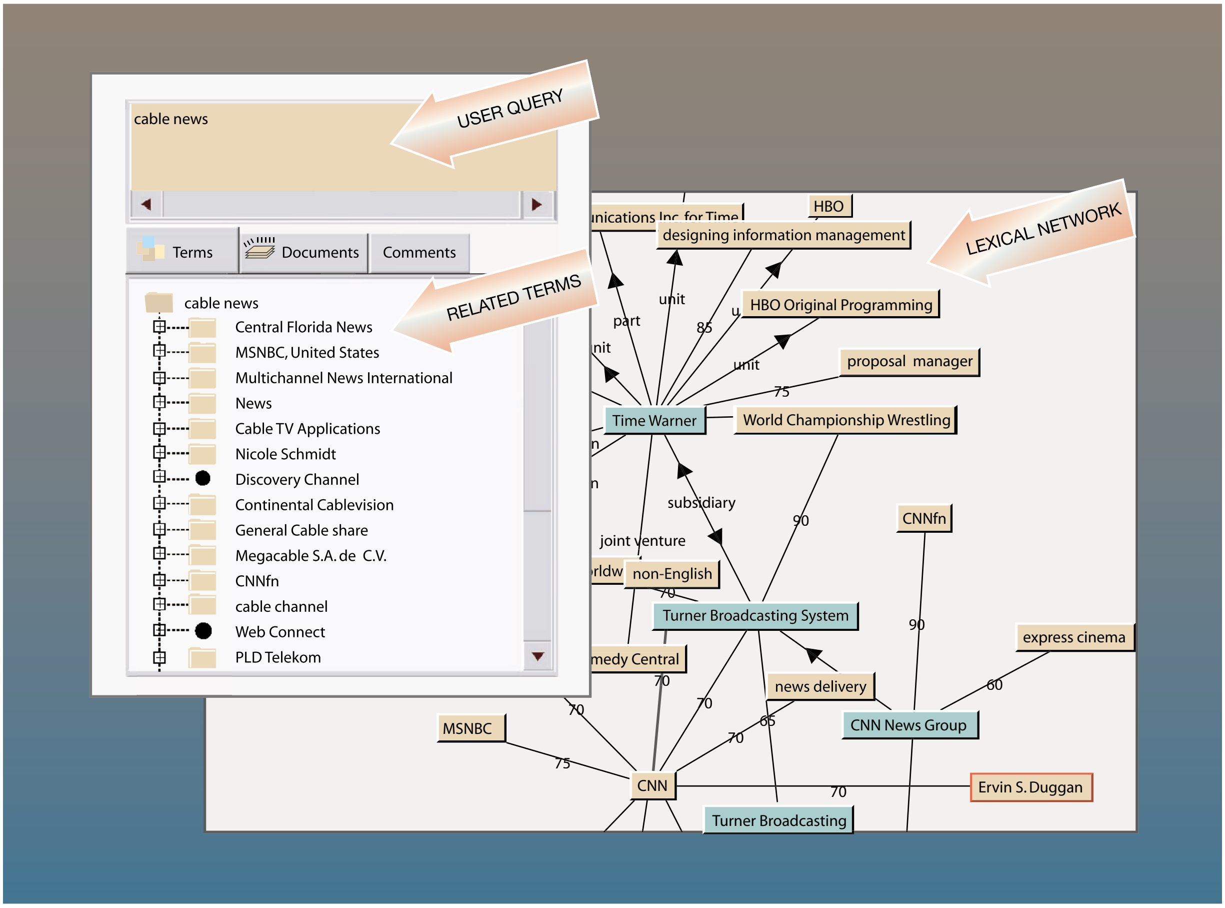
Prompted query refinement (PQR), as the name indicates, is a technique for assisting the user in interactively refining the query, until a satisfactory set of focused and relevant documents is returned. Often, users start with a short and general query, such as the word "Java." When concentrating on their specific information need within their context, they may be (and should be!) unaware of the potential ambiguity of their query terms. (Java could refer to a virus, an island, a type of coffee, or a programming language.) Even if users are aware of this ambiguity, generating the terms necessary to appropriately restrict the query is difficult.[23,24] PQR is a tool that suggests to users candidate terms to add to their queries, as shown in Figure 6. The leftmost object in the figure shows a query, "cable news," and a list of terms related to the query. End users can select one or more of these related terms, and add them to the query specification. Since PQR exploits the features extracted by Textract during the document analysis stage, it only offers terms that actually occur in the collection, in contrast to a general-purpose thesaurus. In Figure 6, many of the related terms are names of cable-related companies described in the documents indexed. (Below we further discuss the lexical network shown on the right.)

During document analysis, the frequency of occurrence of each feature in each document is recorded. A special process combines information about the features and their contexts in the entire collection and creates a special search engine, called a *context thesaurus* (CT). When a user issues a query, it is matched against the CT index, and the hit list returned is not one of document titles but one of terms occurring in documents and ranked by relevance to the query. CT uses an idea inspired by the Phrase Finder.[25] For each feature, it builds and indexes a virtual document, consisting of all the contexts (two to three sentences) in which the feature occurs throughout the collection. When a query matches the virtual document for term X, it is because the query text is sufficiently similar to contexts in which X appears in the collection. The PQR system infers that X is related to the query.

Another advanced search function relevant to portal search is informally referred to as "more documents like this," or more formally as *relevance feedback*. When users find one (or more) relevant documents in the returned hit list, they can submit this feedback to the engine and request to see more such documents. Under the covers, this is achieved by an analysis module (such as Textract, discussed earlier) that extracts salient features from the document selected and turns them into queries that retrieve new documents on the user's behalf. Automatically formulated queries of this kind work very well since they involve feedback from the user. Their other advantage is that they are longer than user-formulated queries and therefore more focused. Finally, they select terms for the new query from the unified context of a single document and therefore reduce ambiguity. In fact, automatically formulated queries have been proven so useful[21] that several search engines now employ them even without user feedback. In a mechanism called "automatic relevance feedback," the engine simply generates and executes queries from the first few documents it returns.

PQR and relevance feedback are two examples of tools that help users find relevant documents through interaction. However, interaction is not always possible. With the increased use of pervasive devices for searching, there is a need to improve search results on the first iteration, particularly the results at the top of the returned list. Recent search algorithms have achieved significant improvements in the search results by ranking documents according to other (nonquery) factors and combining the query-based

Figure 6 Prompted query refinement and lexical navigation of related terms with named relations (e.g., "Time Warner" — "subsidiary (of)" — "Turner Broadcasting System")



and nonquery-based scores. Web pages are ranked high (and called "authority pages") if they are frequently pointed to by other documents. (See References 26 and 27 for many more references to this research.) The success of this method is evidenced by the popularity of the Google Web search service,[28] which first put it into production. Other nonquery-based scores include other measures of document quality, such as number and frequency of accesses and updates and other users' recommendations. A recent example of the use of such extrinsic information to rank documents appears in the system of metrics used in the Lotus Knowledge Discovery Server.

A query is the commonly accepted expression of a user's information need. However, a user may have a focused question in mind that requires a succinct, factual answer. The traditional search paradigm needs to be specialized for this question-answering model. Users ask full natural-language questions, such as "How much does a laptop cost?" Natural language analysis determines the question focus, or the intended answer type, in this case, a price. It also attempts to determine the question goal (shopping as opposed to requesting technical specifications for the machine). Lookup in general and domain-specific ontologies determine the concepts involved (here, specific laptop models). Based on this analysis, the question is translated into a query and processed by the search engine. In this example, "cost" and "how much" are removed; "laptop" is expanded to include synonyms such as ThinkPad*, and a special token (MONEY) is added. To ensure a good match, similar processing is done on the document collection to identify and index semantic concepts (such as monetary amounts) prior to searching. Finally, the ranking algorithm is manipulated to return short passages instead of full documents. Frequency of occurrence is not important, and ranking is determined based on the presence of all query terms in close proximity.

Automatic question answering is a new area of research, combining traditional information retrieval, state-of-the-art natural language processing, and knowledge representation for a deeper understanding of a particular domain. Its coverage is limited at present, but it is driving our research agenda into the next generation of portal technologies.[29]

The search methods we have been describing imply that KWs create and execute search specifications. An alternative is for the system to automatically generate searches on some basis and present results to users. Personalized search methods push information to users based on descriptions of users' interests. For example, users may want to be alerted or notified about new documents related to a customer or product technology they are currently focused on. These interests may be explicitly expressed in profiles created by users, mentioning customers and product topics. Or user interests may be inferred from analyzing documents that KWs browse on the portal[30] or from analyzing e-mail content, or discussion forums for correlations between topics and people who discuss them.[31]

Personalization information can be used in more than one way. In prototype versions of the IBM Global Services K Portal, we extract the categories associated with the documents browsed by the KW and use this information to automatically augment user queries by either restricting the search to these categories or assigning higher weights to documents from those categories.[30] Keywords in browsed documents, or keywords derived from profiles, can also be used to create or augment search specifications. The system can identify other users with similar patterns of usage and can recommend them as members of communities of interest. Personalization can also be based on analyzing query and query results, as done by the knowledge agents advertised by portal vendors such as Plumtree Software.[10] This is an active area of research at the IBM Research Laboratory in Haifa, Israel.[32]

Find: Browsing and navigation. Browsing and navigation are knowledge work activities that go hand in hand with the search function. Since information retrieval is an iterative process, it often consists of a query-based search that returns some initial information, followed by browsing of the contents of the returned hits to learn more about the topic. This action often produces a reformulation of the query, which initiates another search. Since portals are built to assist users with large quantities of information, they need to include summarization tools that extract the most important information from documents and display it to the user. Unlike human-generated abstracts, automatic summaries consist of a collection of sentences (or sentence parts) extracted from the document, with no new text generated. The quality of these excerpts is not as good as human-generated prose—they may seem choppy and are usually not as concise—but they are nevertheless quite useful. There are several kinds of summaries. Longer informative summaries (about 20 to 25 percent of the document length) can capture all the main points of a document. Shorter indicative summaries (one to three sentences long) are usually sufficient for determining whether the document is relevant and should be accessed, read, or translated. Studies have shown that indicative summaries are sufficient for humans to complete tasks without having to read the entire document, thereby saving considerable time and effort.[33] A third kind of summary is query-based summaries. They are typically very short and consist of the most important sentences where the query terms are mentioned. A fourth kind of summarization, keyword summaries, presents KWs with a simple list of technical terms, corresponding to salient names and phrases automatically extracted using an analysis tool such as Textract.

Document summarization works by ranking the sentences in the document for importance and then displaying as many of them as the requested length permits in their original order. The rank of a sentence consists of several factors. One is how many salient textual features it contains, calculated according to the tf/df formula explained earlier, with extra weight given to features that occur in the title and headings. In addition to textual features, the structure of the document also plays an important part, according a higher score to sentences in prime locations (such as document initial or final). For longer summaries, a technique called *topic segmentation* is also used to select summary sentences. This technique examines the distribution of words in the document and identifies break points (at the end of sentences or paragraphs) where the topic changes. Topic shifts are usually marked by a change in the distribution of words, since different words are associated with different topics. To ensure that all topics are covered, the summary includes at least one sentence from each topic segment.[34]

Navigation can be described as a form of browsing, not of a single document, but over a group of documents. We have developed a technique for multidocument summarization (MDS)[35] that blends summarization and navigation. MDS captures the content of a group of related documents, such as the first 100 documents on a search hit list, or the documents in a cluster formed by automatic taxonomy generation. It shows the subtopics that can be identified within the group in various ways: the terms that characterize each subtopic, a few sentences that best represent each subtopic, and the relationship of each document to each subtopic. This categorization allows the user to grasp the different aspects of a topic discussed in the documents without having to read any document in its entirety and to easily navigate from one subtopic to another using a graphical interface. The interface also provides a means to gauge the relative importance of these aspects by examining how many documents are close to each subtopic, and to position the cursor on a document to see at a glance its position with respect to each subtopic.

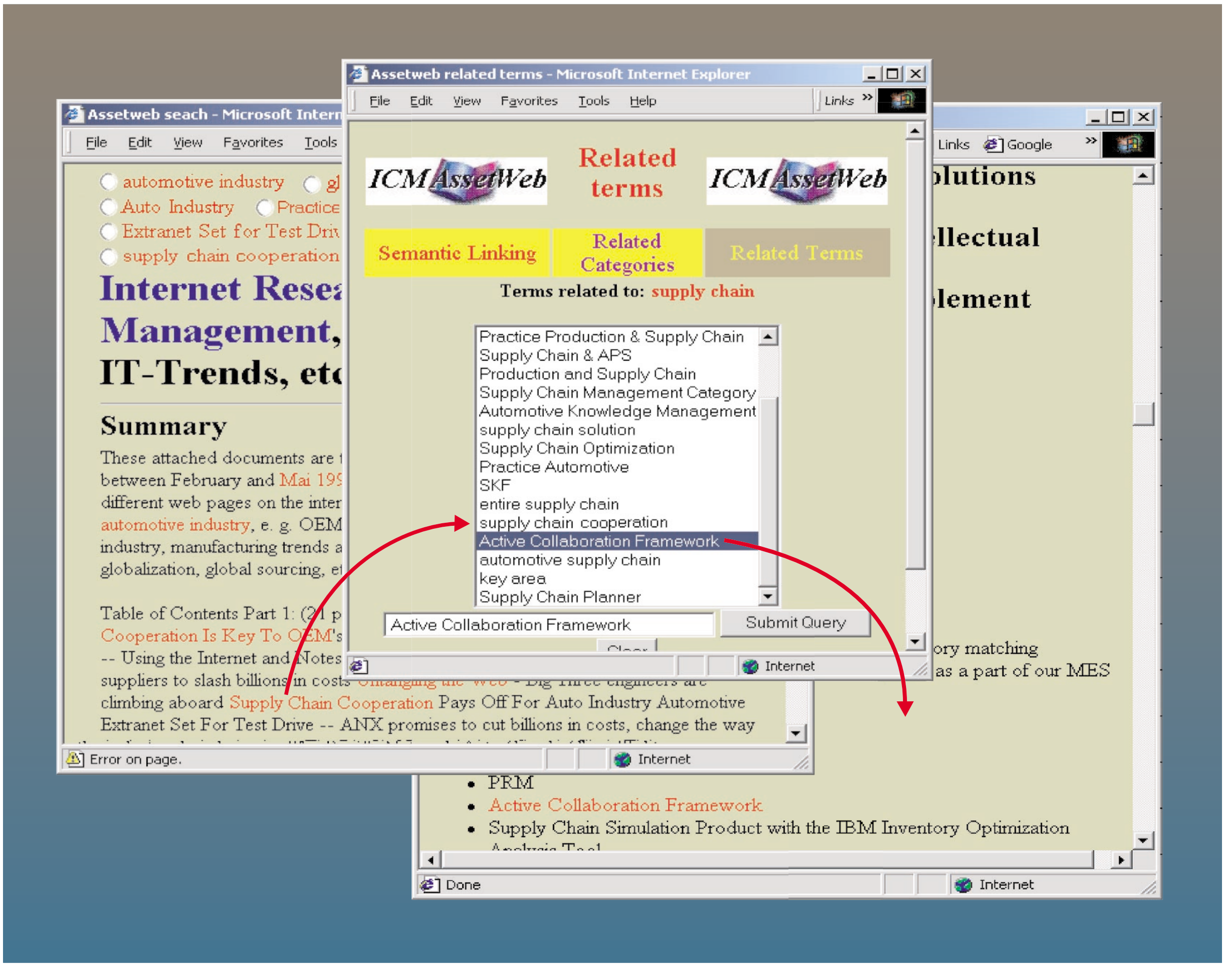
Like browsing, navigation is also complementary to searching. Both methods get the user to information that is relevant. Navigation is partially controlled by the user, who chooses where to go next. It is also constrained to a large extent by the organization of the information in the portal, which is designed by the administrator, as well as by technologies such as categorization, lexical navigation, and active markup, described below.

Category navigation is navigating along the taxonomy that groups documents by categories (as described earlier) and is most closely related to searching. The search function selects a group of documents that resemble the query, whereas category navigation selects a group of documents that resemble each other. Combining the two is very powerful. As users of Web search services such as Yahoo! have all experienced, getting to relevant information is usually the result of interleaving search and category navigation. In the IBM Global Services K portal, this capability is provided so that a user can choose a category first, and then issue a query against the documents in the category. Choosing a category first creates a more homogeneous collection to search within, and therefore can yield more focused results. Even if a category was not preselected, the K portal middleware will allow documents resulting from a search to be organized into the categories to which they belong (analogous to the Northern Light search service[36]). Within each category returned, documents are ranked with respect to the query.

We have also developed a technique called *lexical navigation*[17] to allow users to navigate among salient concepts that have been identified in the collection and represented as textual features. These concepts are linked to one another in two different types of relations. Unnamed relations, based on co-occurrence, indicate that two concepts are related in some unknown way. Named relations,[37] based on linguistic patterns identified in the text, indicate the precise nature of the relationship (e.g., location, kinship, or employment), as described earlier. These concepts and relations form a network, with concepts as nodes and relations as links. Once the user has entered the network, for example, by using the prompted query refinement mechanism to select one or more concepts that are relevant to the query, he or she can then follow relations and navigate to other concepts in an unconstrained way. Figure 6 shows both PQR and an example of a lexical network. We believe that this form of navigation is potentially helpful for the novice, who is trying to become familiar with the scope of a collection of documents. The advantage of the graphical display is that users can focus on a particular neighborhood of interesting terms and easily observe the interconnections among several terms at once. However, when the networks become very large, graph layout can become difficult, and users risk losing intuitions about their location in concept space (see Conklin[38] for a classic analysis of usability issues pertaining to complex hyperlink graphs).

Finally, we put all of these navigation modes together with a technique we have prototyped called *Active Markup*, which links summaries, documents, and concepts.[39] Figure 7 shows an example. When a document is accessed, its short keyword summary appears at the top of the page. Each sentence or initial word in the summary is an active link to the same sentence in the body of the document. Thus, the summary serves as a launching point to the part of the document that is of interest. The keyword summary also supports navigation. Each keyword is a link to other concepts related to it, as well as to other documents containing it. This form of navigation is less structured and more associative by nature than category

Figure 7 Active markup, where automatically extracted technical terms are turned into hyperlinks to related terms and to other documents that contain the same terms


Assetweb seach - Microsoft Intern
File Edit View Favorites Tools
Assetweb related terms - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Links
Google
ICM AssetWeb
Related terms
ICM AssetWeb
Semantic Linking
Related Categories
Related Terms
Submit Query
Clear
Internet
Error on page.
Done

○ automotive industry ○ gl
○ Auto Industry ○ Practice
○ Extranet Set for Test Driv
○ supply chain cooperation

Internet Resea
Management,
IT-Trends, etc

Summary

These attached documents are
between February and Mai 199
different web pages on the inter
automotive industry, e. g. OEM
industry, manufacturing trends a
globalization, global sourcing, et

Table of Contents Part 1: (21 p
Cooperation Is Key To OEM's
-- Using the Internet and Notes
suppliers to slash billions in costs
climbing aboard Supply Chain Cooperation Pays Off For Auto Industry Automotive
Extranet Set For Test Drive -- ANX promises to cut billions in costs, change the way

Terms related to: supply chain

Practice Production & Supply Chain
Supply Chain & APS
Production and Supply Chain
Supply Chain Management Category
Automotive Knowledge Management
supply chain solution
Supply Chain Optimization
Practice Automotive
SKF
entire supply chain
supply chain cooperation
Active Collaboration Framework
automotive supply chain
key area
Supply Chain Planner

Active Collaboration Framework

lutions
llectual
lement

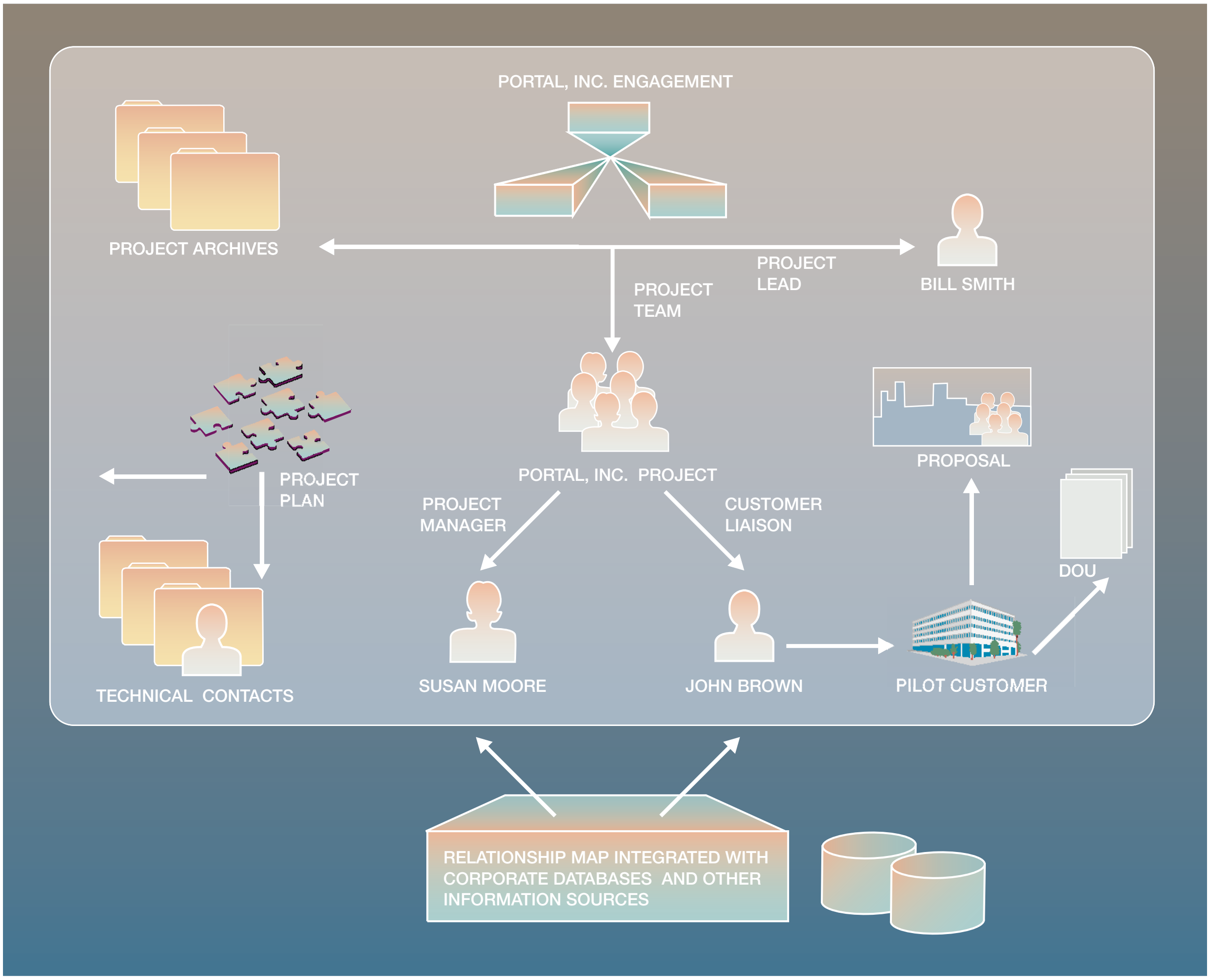
ory matching
as a part of our MES

- PRM
- Active Collaboration Framework
- Supply Chain Simulation Product with the IBM Inventory Optimization Analysis Tool

navigation. It provides a means for browsing the space of documents without having to choose a category or formulate and rephrase queries. We sometimes refer to navigation with active markup as "query-free searching."

Supporting knowledge workers' analysis, synthesis, and authoring of information. Broadly construed, knowledge work involves solving problems. This definition implies human analysis of information, synthesis of new information expressing implications and solutions, and authoring of new artifacts to communicate solutions to colleagues. For example, in a customer engagement, presentations are prepared, proposals and development plans are generated, project teams formed, roles and responsibilities defined and negotiated, budgets developed, and so on. Searching and browsing are a first step, but the information returned needs to be analyzed by KWs in task-oriented ways. Many of the software tools used to support this task of human analysis have been developed outside KM contexts as standard office productivity applications, such as word processors, presentation graphics (e.g., Lotus Freelance Graphics**, Microsoft PowerPoint**), spreadsheets, business graphics, project management software (e.g., Microsoft Project), and document templates that represent forms and outlines for documentation.

Figure 8 Visualizing knowledge resources



In addition to these tools, new tools specialized for KM are emerging for analyzing and synthesizing information. We have already described tools for browsing search results such as multidocument summarization and lexical navigation. Creating such tools is an active area of research in computer science, information retrieval, and cognitive psychology, and much more can be found in Card, Mackinlay, and Shneiderman.[40] These tools exist as research prototypes, and have not yet appeared in commercial applications. Other KM tools such as Grape-VINE[2,41,42] and the former Knowledge X[41] have emerged in the commercial domain. They are intended to help KWs generate relationship maps or graphic visualizations of entities and relationships. Some examples are schematized in Figure 8. These visualizations express organizational structures, connections among people, and project-related topics and artifacts. The goal of these tools is to provide a heterogeneous and open-ended workplace for representing objects and relationships and to help KWs discover potential new relationships. Representations of entities and relations are integrated to some extent with databases containing information describing entities, such as organizational, personnel, and project-related databases.

Project collaboration is another focus in research and commercial domains. An example of the former is TeamSpace,[43] a prototype that provides real-time distributed meeting support using shared workplaces, telephony, and video conferencing, and in addition, the tool archives meetings artifacts such as presentations and video and audio recordings. Users can browse collaboration events on a graphic timeline and select meeting artifacts, such as audio and video records, to browse and play back. Tools like Babble[44,45] enhance real-time chat-like communication among collaboration teams, presenting graphic representations of topic groups, including the identity of discussion participants, and even visual indication of their real-time participation. Issue-Based Information Systems (IBIS) capture team design and problem-solving using text-oriented outlines or visual maps to represent discussion topics and the issues related to them. (Conklin[38] is a somewhat dated, but still definitive discussion of IBIS in the context of hypertext computer systems; Conklin and Begeman[46] discuss a graphic version called gIBIS.) A version of the IBIS system[47] is used by the ICM AssetWeb.

Specialized productivity tools have been developed to support KWs in call centers. These workers, known as customer support representatives (CSRs), need fast access to specialized information as they attempt to identify a solution to a customer problem during a live telephone conversation. The DataCase system, developed at the IBM T. J. Watson Research Center for assisting IBM help-line CSRs, involves manual creation of decision trees that are traversed by the CSR to identify the nature of a technical problem.[41,48] The TAKMI project, at IBM's Tokyo Research Laboratory,[49] classifies and describes customers' problem inquiries in terms of the key concepts they express, using enhancements of the text mining tools discussed earlier. It correlates the textual features extracted from documents with other meta-data features, such as the date, or manufacturing location, to spot trends in customer problems, and the products and features associated with them. These feature correlations can also be shown in a variety of information outlining views,[50] e.g., timelines for trend analysis or event distributions plotted against geographical locations.

These tools emphasize visualization techniques, with some degree of automated generation and update of visualizations, intended to help users discover new facts and implications of information. The rationale for visual techniques is based on the fact that humans are highly visual, and much human reasoning and problem solving is facilitated by visual metaphors and techniques as evidenced by the widespread use of presentation graphic artifacts in office productivity applications and the great care taken in producing them (see Tufte[51] and Card et al.[40] for an extensive review of information visualization techniques). What existing office productivity tools lack is an automatic relation between the representations of entities and the data that they represent. In some cases, this relationship may not be possible to obtain automatically, because too much human intelligence was involved in the synthesis and conceptualization that created it (reflect on the complexity of typical presentation graphic slides). However, in other simpler cases, such as representing simple connections among project team members, customers, and project artifacts, it may be possible to automatically link information about these entities as stored in a database to their visual representations. Such updating still requires a level of information and application integration that is not yet commonplace. Keeping seemingly straightforward artifacts such as on-line Web pages, resumes, and personal information databases current is a difficult task. Moreover, the conceptual structures created by standard office productivity tools are more complicated than relationship maps. These structures require a great effort to create and maintain, and typically require formal human explanation to understand and draw implications from, involving intensive human communication and presentation skills. More innovation is needed to automate the generation and update of these kinds of structures and to support the discovery of implications based on them.

Once KWs have analyzed information and synthesized a solution, they need to communicate it. Several innovations in authoring are emerging. Collaborative authoring allows multiple authors to keep track of multiple contributions, annotate contributions of coauthors, and merge multiple edits. Collaborative annotation allows annotation by readers at large, enriching documents with comments and additional perspectives.[52,53] Robertson and Reese[54] describe a corporate research-desk prototype where research results related to a topic are organized in hyperlinked briefs for reuse in future inquiries related to the same topic, and internal versions of such research briefs are provided as a service through an IBM Global Services research desk organization. Smart documents use a portal-like search to automatically retrieve relevant information for the document at hand. These tools analyze what the author is composing and suggest collateral information that might be of use. They look up references, make sure citations are accurate, and provide example passages from other documents. The SOALAR (Solution Architecture Logic and Reuse) project at the IBM T. J. Watson Research Center[55] enhances a document management system specialized for creating contracts and proposals by retrieving potentially reusable document components from prior documents, in the appropriate contexts. Since the tool is aware of the structure of such documents, it also checks internal consistency and completeness of the current document and captures the resulting new document as a reusable asset back into the repository.

The relevance to portals is at least twofold. First, the artifacts created by these tools contain useful information that can become part of a K Portal, if the crawling and content analysis methods can access and process them, which is not the case today. Second, keeping these tools for analysis, synthesis, and authoring updated with relevant information provided by search and text mining capabilities can make them more useful. We believe that as portals evolve into more broad-based knowledge workplaces, these functions will become increasingly interwoven with other tools that support analysis, authoring, and project execution. The results of searching in a new generation of analysis and synthesis tools will be integrated into targeted information flow and into a broader range of information visualization structures, thereby helping KWs apply their human intelligence to become aware of and discover new relationships among information elements. We discuss the innovations implied by this scenario in the sixth section of this paper.

Distribute, share, and collaborate. The last high-level knowledge work task in Figure 1 is sharing expertise. The *raison d'être* of portals is dissemination of knowledge captured in electronic form. KWs can distribute information by submitting documents to repositories accessed by the portal crawling infrastructure. KM practices may be needed to evaluate the quality of documents and to assign meta-data attributes so that documents can be categorized or handled in a standard way in a portal infrastructure. The progress of a document within some electronic dissemination processes (e.g., certification, category meta-data, and authorization) can be managed by workflow software. Within a project team, project relevant information may also be distributed via shared document repositories such as Lotus Notes TeamRooms, or via electronic mail with attachments. Portals support sharing of documents and collaboration among KWs by giving them access to summaries of persons' resumes and areas of expertise and by publishing documents. KM tools like the ICM AssetWeb exist in a workstation environment that includes tools for collaboration, such as electronic mail, calendar, real-time meeting support with shared applications that are integrated with telephony, instant messaging and awareness,[8,56] and video exchange. Beyond these portal connections, collaboration support is a broad area of research and product technology.[45,57]

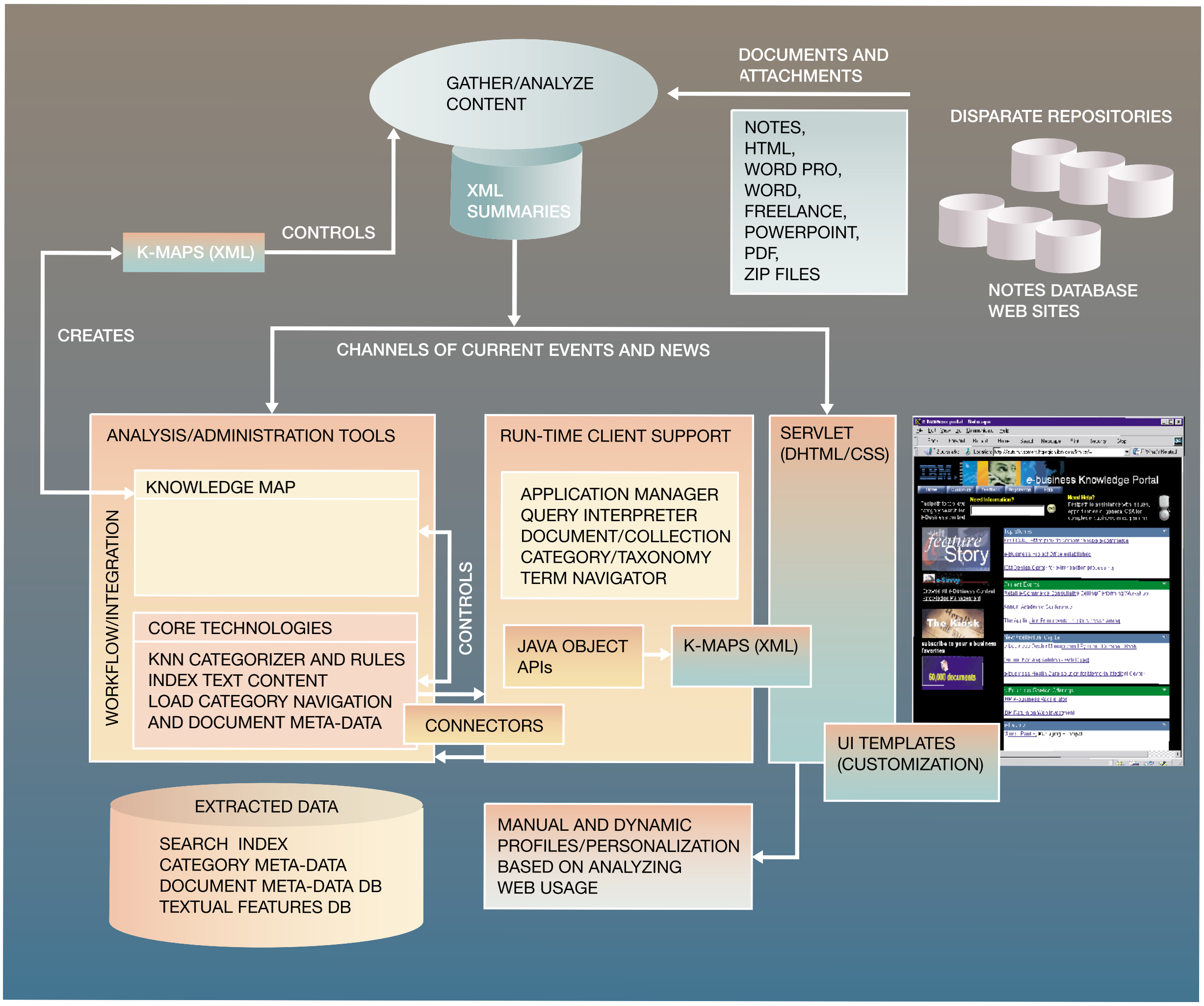
Building and maintaining knowledge portals

Knowledge portal technologies are typically associated with a set of administrative tasks, requiring the exercise of various kinds of expertise. Figure 9 describes a general portal architecture that captures the integration of technologies and human intervention.

Portal application architecture and implementation. Figure 9 indicates the major K Portal components we have developed, beginning along the top of the figure with "Gather/Analyze Content." This crawling component gathers and extracts text and meta-data content from collections of documents distributed in multiple repositories over a network. The extracted content is rendered in a standard XML format, which allows its exploitation by various text analysis and indexing processes, identified in the lower left box in Figure 9. The XML meta-data are loaded into relational database (RDB) tables, as are the category features. The text content of the documents is indexed in a searchable text index, and the documents are automatically categorized. Search and navigation functions in the application client (UI) are based on real-time access of text search engines and RDB tables by a set of run-time classes. If personalization functions are used, such as we described earlier, they may require storage of user profiles and records of portal usage, aggregated for the identification of communities of users.

In our K Portal, we developed a set of object-oriented portal abstractions for run-time support of applications, captured in Java** classes that correspond to familiar entities such as documents, categories, and queries. This middleware provides a higher-level programming interface aimed at improving application development and more powerful ways to manipulate the results of text and RDB searches during run-time processing of user interactions. For example, the application client may allow end users to enter queries using a simpler syntax than the underlying search engine can accept, or it may define sets of default parameters transparent to the user. Ap-

Figure 9 Technology architecture for an example knowledge portal, from gathering document information, to indexing and categorization, and run-time generation of Web pages



plication enabling middleware then parses user queries and transforms them into search specifications appropriate for one or more search engines. A new generation of application enabling middleware will allow results to be merged and manipulated in federated searches across multiple heterogeneous databases. Customizable application clients render backend text and meta-data features flexibly. Typically, the middleware and application client software operate in a rich Web infrastructure that includes a variety of techniques for managing Web access, performance, and generation of Web pages with dynamic data (e.g., search results).

Knowledge workers typically do not have to concern themselves with the implementation and maintenance mechanisms of portals, although they can experience the impact of these on performance and integration of end-user functions. For example, the middleware may play a role in managing user registration and controlling access to documents. The impact on the user is manifested in log-on procedures and document access limitations. Application integration affects how easily new functions can be implemented, how easily code can be customized and modified, and how seamlessly data objects in one tool can be used by another. In our experience, the path from prototype functions to availability in a supported K Portal application can be quite lengthy, in part as a result of these considerations.

Figure 9 (upper left quadrant) also alludes to a Knowledge Map editor (K-Map) tool used to create specifications that control crawling and categorization. Note that the use of the term K-Maps is rapidly becoming a generic term with slightly different meanings in different contexts, e.g., in the IBM Global Services K Portal and the Lotus Knowledge Discovery System** (see Davenport and Prusak[2] for still more uses of the term "knowledge map"). However, most uses refer to the capability of building and editing taxonomies. From a knowledge administrator's point of view, K-Maps are intended to specify what repositories to access for the portal and how to categorize documents. K-Maps are implemented as XML descriptions that can be interpreted by the K Portal indexing, analysis, and categorization programs in order to control their behavior. K-Maps are high-level tools used to create taxonomies. They have the look and feel of a directory navigator (e.g., Microsoft Windows Explorer**), and allow users to drag and drop training documents into subcategories. Other capabilities under development include forms for capturing rules that specify what repositories to crawl, or alternative rule-based methods for categorizing documents.

From a programming viewpoint, K-Maps are intended to facilitate the maintenance of K Portal administration programs with declarative specifications for how to organize information and manage the interoperability of software components. For example, in the IBM K Portal context, K-Maps might be used to control the crawling process (specifying sources and crawling parameters), to control how crawler output is to be used in text indexing and categorization processes, and to specify how users view and navigate taxonomies in the K Portal Web client user interface. K-Maps are a major step toward a new information and software architecture where software components are services that interact in a standard XML protocol, and where a declarative set of attributes represents implied rules for the operation of each service, its required input, and the results it produces. The use of XML is also changing the nature of the analysis processes—text analysis and information extraction are now XML enabled. New search engines are being developed to allow searches on XML structures that represent both textual features and meta-data.[58] As a consequence, component applications do not need to be compiled together, but can interact in simple, standard ways, based on simple Web-based client-server protocols.[59]

Portal management. Ideally, the technology components described in Figure 9 will run virtually automatically, minimizing the role of human management. Although this situation is increasingly the case for many K Portal tasks, there are still aspects of portal operation that require human involvement and oversight. These aspects include managing the process of crawling, indexing, and running categorizers. Other tasks involving content management will likely never be automated. These tasks include, for example, developing and maintaining taxonomies, assessing the quality of search and categorization, and maintaining news channels and highly dynamic sources of information.

Gathering and extracting information requires identifying relevant repositories and specifying crawling rules to gather relevant information and ignore irrelevant information. Web sites and repositories such as Lotus Notes can pose various difficulties to crawling and data extraction. Access rights may have to be negotiated with owners. Dictionaries may need to be defined to map differences in meta-data terminology from one repository to another. Documents may be corrupt, and Web sites may have idiosyncrasies. These problems diminish over time but can be challenging early in deploying portal infrastructures, requiring system administration expertise.

Building or installing a K Portal infrastructure typically requires a range of software engineering skills, such as database and system administration skills and some level of programming where Web clients need to be customized. These administration tasks should be supported by high-level tools. State-of-the-art Web generation software, such as JavaServer Pages**, is also critical to rapid customization and iteration on client user interfaces. Once the K Portal is running, the skills needed are more in line with KW goals and expectations and are less system-related. Domain experts need to develop taxonomies, identify new sources valuable to the community, manage certification, and possibly classify new intellectual capital.

How much quality control should be exercised in accepting assets into the portal repositories is an open question. Better quality requires great effort on the part of a few authors and editors but minimizes the frustration of many end users and maximizes their efficiency. An approach to the issue of varying quality is to create a process for submitting and qualifying documents. This role can be assumed by specialists who are domain experts (the "core teams" mentioned earlier). Although it increases the value of portal assets, quality control can have disadvantages. It can lead to bottlenecks in getting information into the portal repository in a timely manner, which may discourage KWs from both submitting information and using the portal for business-critical decisions. A consequence we have observed is that, in some communities, informal portals, that are exempt from the formal quality control requirements, proliferate. This occurs when portals are built and supported in partisan ways by small organizations.

There is an inherent tension between trying to capture information as quickly and broadly as possible while ensuring its quality. Organizations grapple with this issue when they establish policies for managing quality. We have seen stipulated regulations requiring a practitioner to verify that all the intellectual assets associated with an engagement have been submitted to the portal before the engagement can be closed. Organizational incentives also play a role. Authors may be acknowledged or compensated if their documents are valued by others. (An excellent discussion of these issues can be found in books by Davenport and Prusak[2] and Stewart.[60]) Automation can play some role here. For example, click logs can determine how many times a document is accessed. Documents with links can be analyzed for their connections to and from other documents (as we discussed in an earlier section; see also Chakrabarti et al.[26]). Another promising approach involves algorithms for detecting "useless" documents that do not have much content or value.[61] To identify such documents, we have asked users to skim documents and judge them useful, somewhat useful, or useless. We then used machine learning techniques to train on these documents in order to recognize similar documents, presumed useless, and eliminate them from the repository. This technique works well for very obvious cases, such as documents that contain mostly standard template verbiage with little additional content. Still, it leaves open the issue of subtle quality problems, such as poor style or inaccurate information, as well as the problem of improving and correcting these documents.

In our experience, developing taxonomies and ensuring the accuracy of categorizing documents is a difficult task. We discussed technology for clustering and categorizing documents and the skills needed for this task in an earlier section. Building taxonomies requires a domain expert who understands how users would like the collections organized and what terminology will be intuitive for naming categories. In our experience, domain experts need to understand the users who make up the community (novices and experts alike) and be able to produce a coherent organization of the domain that will be suitable for them. In the IBM Global Services experience, developing taxonomies has turned into a methodology that has become an integral part of internal K Portal deployments. The expert should also know how to use tools, such as K-Maps referred to earlier, which allow easy creation and editing of taxonomies, finding of training documents, and assignment of documents to categories by dragging and dropping. Search tools can help identify training documents, allowing users to search for documents that contain terminology relevant to the taxonomy name or description. These tools assist in the building of the taxonomy but still leave the burden of evaluating its quality to the human.

Tools have been developed to calculate metrics that can help in this evaluation. The e-Classifier tool developed at the IBM Almaden Research Center is a taxonomy-editing application that produces quantitative measures to characterize a categorization scheme. It comes with visualization tools for analyzing the distribution of documents in categories. The user can see, for example, how big each category is, how similar its member documents are to one another, and how well differentiated one category is from another. If a category is too big, or not coherent enough (documents in a category are not similar enough in some sense), the category can be broken into smaller categories, and documents can be distributed appropriately (see References 16, 19, and 20 for an overview of publications on text analysis including categorization and clustering; see also discussion of aspects of this technology in the Lotus Knowledge Discovery System).

A final requirement for portal development is customization and personalization. In addition to the fundamentals of text search and category navigation, portals provide information targeted at a specific user community. For example, bulletin board items (shown in Figure 2) and news items (not shown) can be customized and maintained by staff in a support organization. Some of the customization burden can be alleviated by allowing individuals to personalize their own portal. My Yahoo!, for example, allows users to create their own portal around a core set of Yahoo! functions, with other information and services of interest specified. Beyond news channels, a trend in K Portals is to enable applications to appear in windows within the portal context. Plumtree Software calls these windows *gadgets*. The Lotus K-station** portal application calls these windows *portlets*. An example is a business graphic tool that pops up in the context of a spreadsheet to display numerical data. These miniapplications have great potential in our view for transforming K Portals into more broad-based KM workplaces. The key issue is how well these miniapplications actually integrate with one another in task-relevant ways.

Portal in a box? The administrative and support responsibilities discussed here question the feasibility of a portal "right out of the box," as advertised by some vendors. This ideal is certainly plausible and one that motivates the development of tools such as the K-Map discussed earlier. We should differentiate between easy deployment of the technology and the effort required for integrating it into a heterogeneous software environment and operating it day to day. Even for deployment, the ease of installation should be assessed against the size of the communities, the scope, diversity, and quality of information resources to gather and index, and the skills available for the roles and responsibilities outlined above (e.g., taxonomy development). There is room for more systematic competitive evaluation practices to assess these trade-offs. Proprietary evaluations are often undertaken, but the results are rarely available in the public domain.

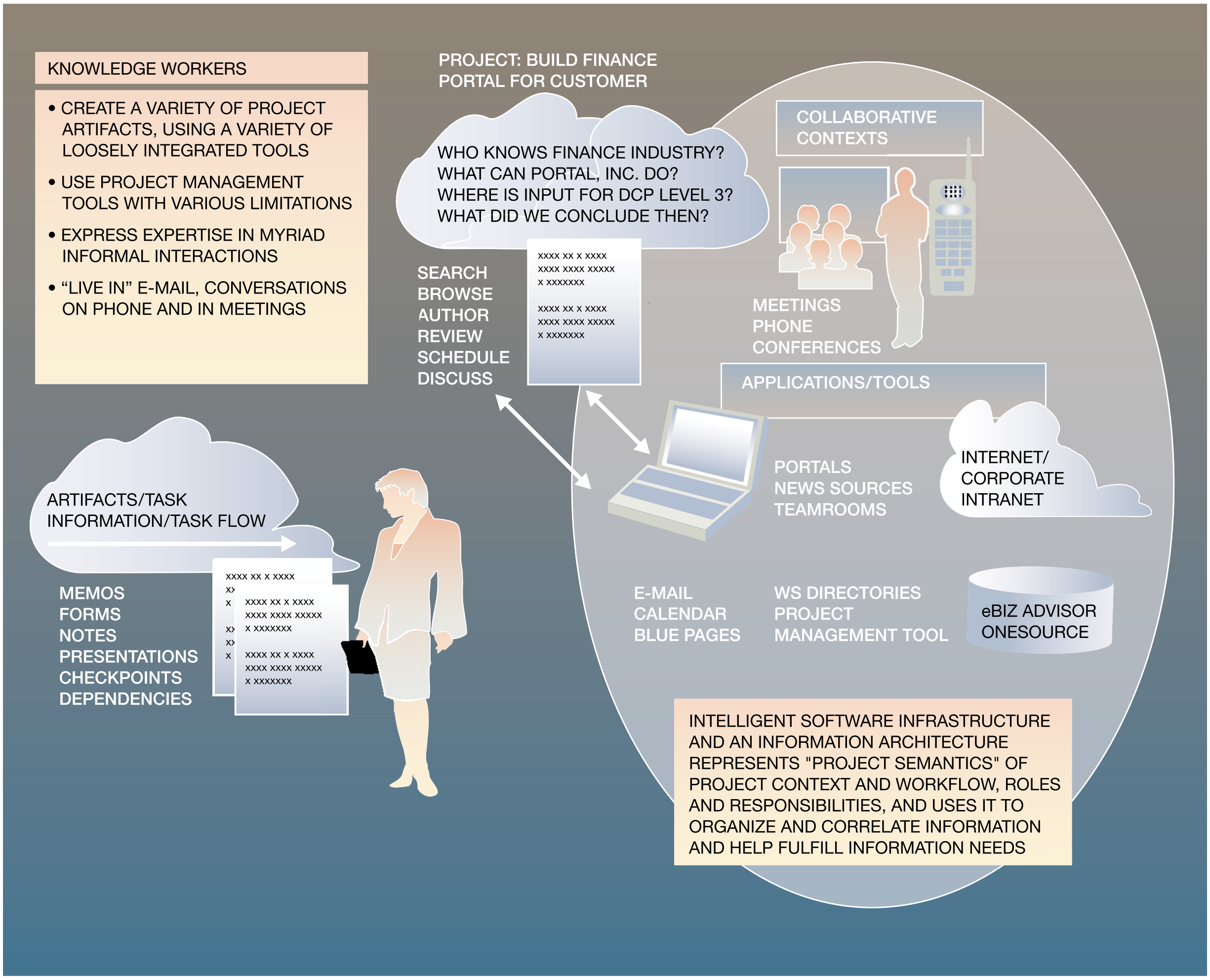
Knowledge portals in an expanding knowledge workplace

Knowledge portals represent a combination of technologies and practices that serve key knowledge work tasks. As we noted, however valuable portals are, they nonetheless represent only a portion of the support KWs need to be effective. Other tasks are not yet well integrated in the broader knowledge workplace. Figure 10 suggests the dimensions of this larger electronic knowledge work context and how K Portals need to evolve. We depict in schematic form the variety of tools and collaboration contexts in which KWs operate (right side of Figure 10). These stand in contrast to the tasks, artifacts, and information needs KWs have as they carry out project tasks (left side of the figure). In our view, the digital knowledge workplace of the future will be driven by a more intelligent and task-oriented infrastructure than the one enabled by current KM technology. This emerging knowledge workplace will support targeted knowledge work tasks more directly and integrally, with reference to specific project roles and responsibilities in a collaborative work environment. This workplace will emerge even as the KW's software environment expands and becomes more distributed and varied.

The expanding knowledge workplace. Knowledge work is increasingly carried on in mobile and pervasive computing environments, in the field or on the road, outside of offices, where KWs must use notebook computers or even handheld devices, and without access to high-bandwidth network connections. These constraints influenced design decisions for the IBM Global Services K Portal. For example, the portal needed to be a lightweight Web application, with no Java servlets, thus limiting the interactive sophistication of applications. To accommodate traveling users with a low-bandwidth connection, we display meta-data indicating document size in bytes and attachments. Techniques such as summarization and question answering will also play an important role in adapting search results for display on small devices. But mobility requirements run much deeper than this, of course. For example, information created or available in one application context (e.g., e-mail) needs to be available in other application contexts (e.g., phone mail or a handheld device). The input and output modalities and properties of these different devices vary widely in physical and human-computer interaction terms. The challenge is to map inputs and outputs appropriately from one device to another, where in some cases, the mapping may entail a deep restructuring of information.[62,63]

Most of the knowledge we have been discussing in this paper is expressed in formal electronic documents that take considerable effort to produce. However, KWs express their expertise in other situations as well, such as in meetings or phone conversations, and these situations provide potential new sources of knowledge and expertise. The effort of producing formal documents often inhibits their creation, since the focus of practitioners' work is on customers, not debriefing. However, if KWs do not always have the time or incentive to compose formal documents, they may be willing to dictate reports using speech capture, or allow conversations on the phone or in meetings to be captured and analyzed using speech recognition. Work done in the IBM T. J. Watson and Almaden Research Centers suggests that speech capture from audio or video recording and text analysis of recognized speech text may provide rich new sources of potential knowledge.[64] Analyzing other

Figure 10 Expanding the knowledge workplace


KNOWLEDGE WORKERS
• CREATE A VARIETY OF PROJECT ARTIFACTS, USING A VARIETY OF LOOSELY INTEGRATED TOOLS
• USE PROJECT MANAGEMENT TOOLS WITH VARIOUS LIMITATIONS
• EXPRESS EXPERTISE IN MYRIAD INFORMAL INTERACTIONS
• "LIVE IN" E-MAIL, CONVERSATIONS ON PHONE AND IN MEETINGS
PROJECT: BUILD FINANCE PORTAL FOR CUSTOMER
WHO KNOWS FINANCE INDUSTRY?
WHAT CAN PORTAL, INC. DO?
WHERE IS INPUT FOR DCP LEVEL 3?
WHAT DID WE CONCLUDE THEN?
COLLABORATIVE CONTEXTS
SEARCH
BROWSE
AUTHOR
REVIEW
SCHEDULE
DISCUSS
MEETINGS
PHONE
CONFERENCES
APPLICATIONS/TOOLS
ARTIFACTS/TASK INFORMATION/TASK FLOW
PORTALS
NEWS SOURCES
TEAMROOMS
INTERNET/ CORPORATE INTRANET
MEMOS
FORMS
NOTES
PRESENTATIONS
CHECKPOINTS
DEPENDENCIES
E-MAIL
CALENDAR
BLUE PAGES
WS DIRECTORIES
PROJECT MANAGEMENT TOOL
eBIZ ADVISOR
ONESOURCE
INTELLIGENT SOFTWARE INFRASTRUCTURE AND AN INFORMATION ARCHITECTURE REPRESENTS "PROJECT SEMANTICS" OF PROJECT CONTEXT AND WORKFLOW, ROLES AND RESPONSIBILITIES, AND USES IT TO ORGANIZE AND CORRELATE INFORMATION AND HELP FULFILL INFORMATION NEEDS

sources, such as discussion groups, e-mail communications, project artifacts stored in project workplaces, and Web usage, can also provide information that can be used to find expertise, identify communities of interest, and find new relevant information.[30,43]

Deep task support: Information architecture and intelligent software infrastructure. KWs who use the portal technologies described earlier find them useful for discovering information relevant to projects, but they also express the need for better integration of these technologies with the other tools that they use in projects. This applies to all the KM-related tools we have been discussing, including project management, collaboration support, information analysis, document generation, and data management. The information they produce needs to be readily usable by other databases or tools. If the motivation for K Portals is unification and integration, it is reasonable to pursue these goals across the full spectrum of KM tools. In this subsection, we sketch the broad outlines of the knowledge workplace we see emerging with such integration.

If we elaborated on the "day-in-the-life" scenario in Table 1 to describe in more depth the pragmatic and temporal structure of project tasks, the roles and responsibilities implied by them, information flow, and dependencies, we would note the big gaps in the electronic workflow that supports them. These gaps are too often "left to the user to integrate,"[65] that is, filled by more or less *ad hoc* practices, workarounds, and, in our opinion, nonoptimal, nonelectronic, integration activities developed by KWs. Although current project management tools help KWs represent project execution, the resulting representation is not itself an active driver of application information flow or task execution. Providing deeper, more integrated support for KW projects will require technical innovation in information architecture, that is, in the representation of the structure, and attributes and roles of information associated with tasks. Exploiting a richer, broader information architecture also implies intelligent software infrastructure: software processes and tools that can act on the representation of task information and processes and support and even automate aspects of task execution for KWs.

Currently, information architecture of tasks is implied in the activities and expertise of KWs but not tightly coupled to software architecture. For example, the information architecture we developed in an early prototype of the portal described in Figures 1 and 2 was intended to categorize documents in terms of the processes and subtasks involved in "engaging customers." However, this relates only indirectly to the temporal and pragmatic steps of engaging with customers, and it is dependent on the skill of KWs to understand and exploit. There is no automated workflow implied in this category representation, no constraints or structure to guide KWs in project fulfillment, no explicit representation of milestones or ordering of subtasks, and little relation to the tools and applications used to fulfill elements of the project over time. What is missing is active project support: This means automatic or semiautomatic accessing, organizing, and presentation of project information within the temporal context of day-to-day project management. For example, it would be useful to show a KW who is responsible for summarizing a customer visit, similar reports written by others for customers with similar size, industry, and deliverables. In addition, it would be useful to access and present these similar reports automatically, based on an understanding of the KW project role, and where he or she is in a temporal sequence of project tasks.

The information architecture necessary (but not sufficient) for enabling active task support is a richer representation of task structure, represented, for example, in terms of scripts or schemata.[66–68] These structures are intended to represent structural elements of tasks organized in time, including project roles and responsibilities, information dependencies and flow, and conventions for handling human interactions, including exceptions and breakdowns in communication (e.g., see Flores et al.[69] and Medina-Mora et al.[70]). We envision a new form of K-Maps to represent not only content-based taxonomies, but richer, task-based workflow. End users would experience the value of this architecture because it would enable them to operate in more task-oriented terms, using active project templates.

The smart documents project mentioned earlier provides an example. Currently, IBM Global Services KWs work on documents called *statements of work* (SOW) in the isolated context of a word processor, searching for relevant planning information in a separate portal context, and then using some manual effort to extract relevant information and incorporate it into the SOW. In contrast, a smart document would enable KWs to initiate a context-sensitive search from within segments of the SOW, returning similar segments from other SOWs, based on the attributes of the SOW segment. Carrying this notion further, we can envision other smart project management tools that embody the engagement life cycle in an active way. For example, a new project might start as a timeline view (as in TeamSpace[43] or based on an enhanced project management tool), with links from the timeline to appropriate templates for the documents and analyses that need to be created at each point in the life cycle. Agents would analyze, infer, and gather information that is contextually relevant for each subtask of a project. An approximation of this process is suggested by automated, profile-based search and integration techniques for creating personalized news[71] or topic-related reports.[72] Automatic search and information integration could fulfill the promise of the visualization tools discussed earlier. It would entail more effectively channeling information into visual representations, expanding the kinds of structures and relations that can be captured and expressed, and expanding the kind of implications that can be made. Collaboration support could be enhanced by this infrastructure as well. Meeting mining projects at the IBM Watson and Almaden Research Centers[64] are exploring how to retrieve information dynamically during meetings by using search agents that listen to discussions as they occur. Active Calendar, a project at the IBM Almaden Research Center, is exploring an intelligent method for correlating information in calendars, profiles, and e-mail so that implications related to time management can be drawn for KWs. For example, a

KW who schedules a business trip to San Francisco may be notified of local events of personal or business interest occurring during the visit period.[73]

These scenarios exploit a simple but well-known cognitive psychological principle, namely, that it is easier to recognize and select information than it is to recall and generate it.[74] (This is why, for example, menu-based user interfaces are more likely to be usable than command-based interfaces.) KWs would reuse and manipulate relevant information, adding only what is uniquely required for the specific task at hand. Portal-like functions of search and analysis would be distributed and interwoven throughout electronic workflows and within artifacts and involved in the context of creation and editing.

The automation or semiautomation of these tasks is often based on intelligent agents, which are programs that run autonomously in some electronic environment (such as the Internet), monitor events, respond to them, and carry out actions, based on rules defined over representations of some task domain.[75] Intelligent agents also raise the question of the general role of artificial intelligence (AI) in replacing or augmenting human intelligence in knowledge work. The role of AI is a complicated story, with successes in specialized domains where expertise can be expressed in rules, e.g., medical diagnosis or mineral prospecting. Examples of promising research on intelligent support of cooperative work[76–78] suggest potentially broader KM roles for AI technology. However, in our opinion, AI has not yet provided a reliable, broad-based foundation for intelligent information or software architectures in the domain of KM technology (see Davenport and Prusak[2] and Smith and Farquar[79] for more discussion).

Methods of knowledge capture will have to evolve as well as richer representational schemes for describing tasks in rules and schemata and for capturing the flexibility characteristic of human problem-solving and planning. Structure and constraints in task objects and processes need to be balanced against flexibility and open-endedness required for creative organization of new tasks and information objects. In the real world, tasks do not always unfold in a linear sequence. Task flows must allow for activities to proceed in parallel, reflecting activities by multiple participants, and they should also allow for unanticipated difficulties. Although temporal ordering and ordering by information dependency are important, contingencies are needed to reflect breakdowns and repairs in executing goals and subgoals. The knowledge workplace might best be viewed as an active project management tool enhanced with routing, task lists, approval management, and flexible access control to allow participation by people with different roles, including project members, as well as customers and partners.

Broad-based KM product platforms, such as the Lotus Discovery Server and content-management components of the IBM WebSphere Portal Server, are beginning to provide middleware and application development tools that can be used to build workflow processes moving in these directions. However, in our view, more innovation is needed to develop an information architecture and intelligent infrastructure as we have defined them. The capabilities of existing portal and KM tools and applications need to be organized in a more fine-grained and modular Web services-oriented architecture,[5,59] transforming monolithic applications into systems of lighter-weight Web-based components, organized around electronic workflows, and interoperating via lightweight information exchange protocols. In this context, portal functions likewise become distributed components of a larger and more task-oriented electronic workplace.

Ensuring quality and effectiveness of knowledge portals

In various parts of this paper, we have appealed to anecdotal evidence for various claims about the usefulness of KM technologies. The fundamental motivation for KM technologies such as knowledge portals is to make KWs more productive. How can we find hard evidence for claims about utility and usability? We have also suggested that understanding knowledge work will be the driver of more intelligent software infrastructures. Can we discover how KWs use KM technology so that we can capture it in models and rules?

Human-centered evaluation and analysis practices are as important to developing and innovating KM technology as are the hard disciplines of computer science and software engineering. Professional research and applied disciplines in usability engineering, human factors, and human-computer interaction draw on cognitive and behavioral sciences, including anthropology, to provide behavioral methods for analyzing knowledge work, identifying requirements, evaluating system use, and analyzing feedback to guide design. Examples of organizations that do this work are the ACM (Association for Computing Machinery) Special Interest Groups, including SIGCHI for computer-human interaction, SIGIR for information retrieval and text analysis, SIGGROUP for computer-supported collaborative work, or groupware, and SIGKDD for data mining and KM-related research. User-centered design methodologies can be found in Helander[80] and in the contextual design methodology of Beyer and Holtzblatt.[81] Case studies can be found in KM-related areas such as collaboration.[82,83]

A research agenda

The emerging knowledge workplace as we have described it will be driven by a co-evolution of three research initiatives: an evolving understanding of how KWs accomplish tasks, technical innovation in component technologies, and innovation in application integration linking tasks and technology.

First, understanding KW tasks and needs means applying human-centered behavioral methods to discover and analyze knowledge work and to guide the development of new technology with user requirements and user-centered design and evaluation methods. Second, technological innovation at the level of the text-based and search-based components is required. The research challenges continue to be improving how KWs find relevant information, assisting them in seeing implications and connections of seemingly unrelated facts, and helping them accomplish elements of project plans and artifacts. We have suggested that a prerequisite for these innovations is deeper understanding of the semantics of document content and structure and of project task structure.

Finally, a better understanding of knowledge work practices can also contribute to the third area we believe is essential for effective KM tools, which is how the subsystems and components implied by innovation can be integrated into a cohesive knowledge workplace. The challenge is for all the tools to work together smoothly and efficiently, with the ultimate goal of making knowledge workers more effective and productive.

Acknowledgments

Much of the work described in this paper was developed with numerous colleagues. We would like to especially acknowledge our partners in IBM Global Services, notably Ko-Yang Wang and Charles (Rod) Cowan, who have supported collaboration with Research for several years, and who provided many insightful discussions over the years about topics discussed. In particular, our discussion of "knowledge workplace" owes much to Rod's and Ko-Yang's own initiatives for supporting the internal IBM Global Services community. We also thank several colleagues in IBM Research, including Wayne Niblack, Eric Brown, Edward So, James Cooper, Mary Neff, Bran Boguraev, Herb Chong, Aaron Kershenbaum, Anni Coden, and John Prager, with whom we have collaborated on the technologies discussed in this paper (some deployed in the portals described). In particular, Ed So, Eric Brown, and Wayne Niblack were key contributors to the IBM Global Services knowledge portal discussed in this paper. Finally, we thank James Cooper, Alan Marwick, Rod Cowan, and three anonymous reviewers for providing comments that improved this paper.

*Trademark or registered trademark of International Business Machines Corporation.

**Trademark or registered trademark of Lotus Development Corporation, Dow Jones & Company, Inc., Microsoft Corporation, or Sun Microsystems, Inc.

Cited references

1. L. Prusak, "Introduction to Knowledge in Organizations," L. Prusak, Editor, *Knowledge in Organizations*, Butterworth-Heinemann, Boston, MA (1997).
2. T. H. Davenport and L. Prusak, *Working Knowledge: How Organizations Manage What They Know*, Harvard Business School Press, Boston, MA (1998).
3. A. D. Marwick, "Knowledge Management Technology," *IBM Systems Journal* **40**, No. 4, 814–830 (2001, this issue).
4. Yahoo!, Yahoo, Inc., http://www.yahoo.com (2001).
5. M. Carroll, "Beneath the Vortals," *Web Techniques*, 59–65 (February 2000).
6. K.-T. Huang, "Capitalizing on Intellectual Assets," *IBM Systems Journal* **37**, No. 4, 570–582 (1998).
7. IBM Global Services Wins GIGA Global Excellence Award for Second Straight Year, IBM Corporation, http://www-4.ibm.com/software/data/knowledge/download/GIGApr.htm (2001).
8. Knowledge Management: The Mind of Many, Lotus Development Corporation, http://www.lotus.com/home.nsf/welcome/km (2001).
9. Knowledge Discovery Server, Lotus Development Corporation, http://www.lotus.com/home.nsf/welcome/discoveryserver (2001).
10. *Corporate Portals: A Simple View of a Complex World*, digital White Paper, Plumtree Software, Inc., available from http://www.plumtree.com (2001).
11. Autonomy Corporation, http://www.autonomy.com/autonomy/ (2001).
12. Grand Central Station, http://www.research.ibm.com/topics/popups/smart/network/html/gcs.html, Almaden Research Center, IBM Corporation (2001); (also http://www.research.ibm.com/resources/magazine/1997/issue_3/grandcentral397.html).

13. Garlic Project, Almaden Research Center, IBM Corporation, http://www.almaden.ibm.com/cs/garlic/ (2001).
14. E. Brown, "Parallel and Distributed IR," Chapter 9, in *Modern Information Retrieval*, R. Baeza-Yates and B. Ribeiro-Neto, Editors, ACM Press, New York (1999), pp. 229–256.
15. Intelligent Miner for Text, IBM Corporation, http://www-4.ibm.com/software/data/iminer/fortext/downloads.html (2001).
16. Talent: Text Analysis and Language Engineering Tools, IBM Corporation, http://www.research.ibm.com/irgroup/talent.html (2001).
17. J. W. Cooper and R. J. Byrd, "Lexical Navigation: Visually Prompted Query Expansion and Refinement," *Proceedings of the 2nd ACM International Conference on Digital Libraries*, Philadelphia, PA (July 25–28, 1997), pp. 237–246.
18. Y. Ravin and N. Wachholder, *Extracting Names from Natural Language Text*, Research Report 20338, IBM Corporation (1996); also available at: http://www.research.ibm.com/people/r/ravin.
19. "A New Computer Program Classifies Documents Automatically," Almaden Research Center, IBM Corporation, http://www.research.ibm.com/resources/magazine/2000/number_2/selection200.html (2001).
20. Text and Information, Almaden Research Center, IBM Corporation, http://www.almaden.ibm.com/cs/k53/ir.html (2001).
21. R. Baeza-Yates and B. Ribeiro-Neto, Editors, *Modern Information Retrieval*, ACM Press, New York (1999).
22. *An Introduction to Modern Information Retrieval*, G. Salton and M. McGill, Editors, McGraw-Hill, New York (1993).
23. G. W. Furnas, T. K. Landauer, L. M. Gomez, and S. T. Dumais, "The Vocabulary Problem in Human-System Communication," *Communications of the ACM* **30**, No. 11, 964–971 (November 1987).
24. S. Dumais, "Textual Information Retrieval," *Handbook of Human-Computer Interaction*, M. Helander, Editor, Elsevier Science Publishers, North-Holland (1988), pp. 673–700.
25. Y. Jing and W. B. Croft, "An Association Thesaurus for Information Retrieval," *Proceedings of RIAO'94* (Recherche d'Informations Assistee par Ordinateur) (1994), pp. 146–160.
26. S. Chakrabarti, B. E. Dom, S. R. Kumar, P. Raghavan, S. Rajagopalan, A. Tomkins, D. Gibson, and J. Kleinberg, "Mining the Web's Link Structure," *Computer* **32**, No. 8, 60–67 (August 1999).
27. C. Boyer, "Building Community Online," *IBM Research* **36**, No. 4 (1998); also at http://www.research.ibm.com/resources/magazine/1998/issue_4/babble498.html (2001).
28. Google, Inc., http://www.google.com.
29. J. Prager, E. Brown, A. Coden, and D. Radev, "Question-Answering by Predictive Annotation," *Proceedings of SIGIR 2000*, Athens, Greece (2000).
30. C. Aggarwal, J. Wolf, K.-L. Wu, and P. Yu, "The Intelligent Recommendation Analyzer," *IEEE ICDCS 2000 Workshop on Knowledge Discovery and Data Mining* (2000).
31. M. Ackerman and T. Malone, "Answer Garden: A Tool for Growing Organizational Memory," *Proceedings of the ACM Conference on Office Information Systems* (1990), pp. 31–39.
32. Y. Aridor, D. Carmel, R. Lempel, A. Soffer, and Y. Maarek, "Knowledge Agents on the Web," unpublished manuscript, IBM Corporation (2000).
33. T. F. Hand, "A Proposal for Task Based Evaluation of Text Summarization Systems," *Intelligent Scalable Text Summarization Proceedings of a Workshop Sponsored by the Association of Computational Linguistics* (July 1997), pp. 31–38.
34. B. K. Boguraev and M. S. Neff, "Lexical Cohesion, Discourse Segmentation and Document Summarization," *Proceedings of RIAO'2000*, (Recherche d'Informations Assistee par Ordinateur), Paris (April 12–14, 2000).
35. R. Y. Ando, B. Boguraev, R. Byrd, and M. Neff, "Multidocument Summarization by Visualizing Topic Content," *Proceedings of ANLP/NAACL Workshop on Automatic Summarization* (2000), pp. 79–88.
36. Northern Light, Northern Light Technology, Inc., http://www.northernlight.com.
37. R. Byrd and Y. Ravin, "Identifying and Extracting Relations in Text," *NLDB'99 Conference*, Klagenfurt, Austria (1999).
38. J. Conklin, "Hypertext: An Introduction and Survey," *Computer-Supported Cooperative Work: A Book of Readings*, I. Grief, Editor, Morgan Kaufmann Publishers, San Mateo, CA (1988), pp. 423–475.
39. M. S. Neff and J. W. Cooper, "ASHRAM: Active Summarization and Markup" (abstract), *Proceedings of the 32nd Annual Hawaii International Conference on Systems Science*, Maui, HI (1999), p. 83.
40. S. Card, J. Mackinlay, and B. Shneiderman, *Readings in Information Visualization: Using Vision to Think*, Morgan Kaufmann Publishers, San Francisco, CA (1999).
41. G. E. Bock, "Knowledge Management Frameworks," *Patricia Seybold's Workgroup Computing Report* **20**, No. 2 (February 1997).
42. D. Tkach, *IKM Technology: Knowledge Representation Effectiveness*, unpublished White Paper, IBM Corporation (October 7, 1998).
43. W. Geyer, S. Daijavad, T. Frauenhofer, H. Richter, K. Truong, L. Fuchs, and S. Poltrock, "Virtual Meeting Support in TeamSpace," Demonstration at *CSCW'2000, ACM Conference on Computer-Supported Cooperative Work*, Philadelphia, PA (December 2–6, 2000); see also IBM Research Web site at http://www.research.ibm.com/teamspace.
44. T. Erickson, D. N. Smith, W. A. Kellogg, M. R. Laff, J. T. Richards, and E. Bradner, "Socially Translucent Systems: Social Proxies, Persistent Conversation, and the Design of 'Babble'," *Proceedings of Computer-Human Interaction 99*, Pittsburgh, PA (May 15–20, 1999), pp. 72–79.
45. J. C. Thomas, W. A. Kellogg, and T. Erickson, "The Knowledge Management Puzzle: Human and Social Factors in Knowledge Management," *IBM Systems Journal* **40**, No. 4, 863–884 (2001, this issue).
46. J. Conklin and M. L. Begeman, "gIBIS: A Hypertext Tool for Exploratory Policy Discussion," *Proceedings of the Conference on Computer-Supported Cooperative Work* (*CSCW'88*) (1988), pp. 140–152.
47. Corporate Portals: Case Studies and White Papers, IBM Corporation, http://www-4.ibm.com/software/data/knowledge/corporate/details.html (2001).
48. S. Hantler, personal e-mail communication (February 14, 2001).
49. T. Nasukawa and T. Nagano, "Text Analysis and Knowledge Mining System," *IBM Systems Journal* **40**, No. 4, 967–984 (2001, this issue).
50. M. Morohashi, K. Takeda, H. Nomiyama, and H. Maruyama, "Information Outlining—Filling the Gap Between Visualization and Navigation in Digital Libraries," *Proceedings of the International Symposium on Digital Libraries* (1995), pp. 151–158.
51. E. Tufte, *Visual Explanations: Images and Quantities, Evidence and Narrative*, Graphics Press, Cheshire, CT (1997).
52. J. Cadiz, A. Gupta, and J. Grudin, "Using Web Annotations for Asynchronous Collaboration Around Documents," *CSCW'2000, ACM Conference on Computer-Supported Cooperative Work*, Philadelphia, PA (December 2–6, 2000).

53. I. Ovsiannokov, M. Arbib, and T. McNeill, "Annotation Technology," *International Journal of Human-Computer Studies* **50**, 329–362 (1999).
54. S. Robertson and K. Reese, "A Virtual Library for Building Community and Sharing Knowledge," *International Journal of Human-Computer Studies* **51**, 663–685 (1999).
55. D. Ferrucci, "Knowledge Management Through Interactive Document Configuration," *The Second International Conference and Exhibition on the Practical Application of Knowledge Management* (*PAKeM99*), London (April 21–April 23, 1999); see also http://www.practical-applications.co.uk/PAKeM99/.
56. Sametime–Real-Time Collaboration That's Fit for Business, Lotus Development Corporation, http://www.lotus.com/home.nsf/welcome/sametime.
57. *Computer-Supported Cooperative Work: A Book of Readings*, I. Grief, Editor, Morgan Kaufmann Publishers, San Mateo, CA (1988).
58. D. Carmel, Y. Maarek, and A. Soffer, "XML and Information Retrieval: A SIGIR 2000 Workshop," *ACM Conference on Information Retrieval*, *SIGIR 2000* (July 28, 2000).
59. D. Platt, "Web Services: Building Reusable Web Components with SOAP and ASP.NET," *MSDN Magazine*, 100–114 (February 2001).
60. T. A. Stewart, *Intellectual Capital: The New Wealth of Organizations*, Currency and Doubleday Publishing, New York (1997).
61. J. W. Cooper and J. M. Prager, "Anti-Serendipity: Finding Useless Documents and Similar Documents" (abstract), *Proceedings of the 33rd Hawaii International Conference on Systems Science*, Maui, HI (2000), p. 57.
62. C. Schmandt, N. Marmasse, S. Marti, N. Sawhney, and S. Wheeler, "Everywhere Messaging," *IBM Systems Journal* **39**, Nos. 3&4, 660–677 (2000).
63. M. Viveros and D. Wood, unpublished project reports on mobility, knowledge management, and collaboration, IBM Corporation.
64. E. W. Brown, S. Srinivasan, A. Coden, D. Ponceleon, J. W. Cooper, and A. Amir, "Toward Speech as a Knowledge Resource," *IBM Systems Journal* **40**, No. 4, 985–1001 (2001, this issue).
65. R. Cowan, internal presentation, IBM Corporation (April 17, 2001).
66. R. Schank and H. Abelson, *Scripts, Plans, Goals and Understanding*, Erlbaum, Hillsdale, NJ (1977).
67. J. Sowa, *Conceptual Structures: Information Processing in Mind and Machine*, Addison-Wesley Publishing Co., Reading, MA (1984).
68. A. S. Gordon, "The Representational Requirements of Strategic Planning," *Fifth Symposium on Logical Formalization of Commonsense Reasoning*, New York University, New York (May 20–22, 2001).
69. F. Flores, M. Graves, B. Hartfield, and T. Winograd, "Computer Systems and the Design of Organizational Interaction," *ACM Transactions on Office Information Systems* **6**, No. 2, 153–172 (April 1988).
70. R. Medina-Mora, T. Winograd, R. Flores, and F. Flores, "The Action Workflow Approach to Workflow Management Technology," *ACM 1992 Conference on Computer-Supported Cooperative Work*, Toronto (October 31–November 4, 1992), pp. 281–297.
71. S. Elo Dean and L. Weitzman, "SuperNews: Multiple Feeds for Multiple Views," *IBM Systems Journal* **39**, Nos. 3&4, 633–645 (2000).
72. Atomica Web service, Atomica Corporation, http://www.atomica.com (2001).
73. Active Calendar, Almaden Research Center, IBM Corporation, http://www.almaden.ibm.com/cs/activecal.html (2001).
74. B. Shneiderman, *User Interface Design: Strategies for Effective Human-Computer Interaction*, Addison-Wesley Publishing Co., Reading, MA (1999).
75. P. Maes, "Agents That Reduce Work and Information Overload," *Readings in Human-Computer Interaction: Toward the Year 2000*, 2nd Edition, R. M. Baecker, J. Grudin, W. A. S. Buxton, and S. Greenberg, Editors, Morgan Kaufmann Publishers, San Francisco, CA (1995), pp. 811–821.
76. K. Abbot and S. Sarin, "Experience with Workflow Management: Issues for the Next Generation," *Proceedings of the ACM Conference on Computer-Supported Cooperative Work*, Chapel Hill, NC (October 22–26, 1994), pp. 113–120.
77. K. Lai, T. Malone, and K.-C. Yu, "Object Lens: A Spreadsheet for Cooperative Work," *ACM Transactions on Office Information Systems* **6**, No. 4, 1–26 (1990).
78. T. Winograd, "A Language/Action Perspective on the Design of Cooperative Work," *Human Computer Interaction* **3**, 3–30 (1987).
79. R. G. Smith and A. Farquhar, "The Road Ahead for Knowledge Management: An AI Perspective," *AI Magazine* (American Association for Artificial Intelligence), 17–40 (Winter 2000).
80. *Handbook of Human-Computer Interaction*, M. Helander, Editor, Elsevier Science Publishers, North-Holland (1988).
81. H. Beyer and K. Holtzblatt, *Contextual Design: A Customer-Centered Approach to System Design*, Morgan Kaufmann Publishers, San Francisco, CA (1997).
82. J. Grudin, "Groupware and Social Dynamics: Eight Challenges for Developers," *Readings in Human-Computer Interaction: Toward the Year 2000*, 2nd Edition, R. M. Baecker, J. Grudin, W. A. S. Buxton, and S. Greenberg, Editors, Morgan Kaufmann Publishers, San Francisco, CA (1995), pp. 762–774.
83. R. M. Baecker, D. Nastos, I. Posner, and K. Mawby, "The User-Centered Iterative Design of Collaborative Writing Software," *Readings in Human-Computer Interaction: Toward the Year 2000*, 2nd Edition, R. M. Baecker, J. Grudin, W. A. S. Buxton, and S. Greenberg, Editors, Morgan Kaufmann Publishers, San Francisco, CA (1995), pp. 775–782.

Accepted for publication June 1, 2001.

Robert Mack *IBM Research Division, Thomas J. Watson Research Center, P.O. Box 704, Yorktown Heights, New York 10598 (electronic mail: robertmack@us.ibm.com).* Dr. Mack is manager of the Information Design and Access Group. He joined IBM Research in 1981 in a postdoctoral position, and then as a research staff member, after completing his Ph.D. in cognitive and experimental psychology at The University of Michigan in 1981. He received a B.A. degree in physics from Oakland University and an M.A. degree in experimental psychology from Michigan State University. He has worked on a wide range of software systems, both as a user interface and human-computer interaction specialist, and as a project lead for prototype systems and middleware for applications relating to digital libraries, digital video archive and search, and knowledge portals. All of these projects have involved collaboration with and technology transfer to the IBM Software Group or IBM Global Services. Dr. Mack and his team are currently working on projects related to using user task context to enhance the search function, and applying text mining to support knowledge discovery in the life sciences.

Yael Ravin *IBM Research Division, Thomas J. Watson Research Center, P.O. Box 704, Yorktown Heights, New York 10598 (electronic mail: ravin@us.ibm.com).* Dr. Ravin is manager of the Knowledge Structures Group. She received a B.A. degree in English literature and philosophy from the Hebrew University in Jerusalem (*cum laude*), an M.A. degree in teaching English as a second language from Teachers' College, Columbia University, and a Ph.D. degree in linguistics from the Graduate Center of the City University of New York. Since then, she has been working as a research staff member at the T. J. Watson Research Center on a variety of natural language processing and information retrieval projects. Dr. Ravin created the IBM named-entity recognizer and led its development and transfer into a product (Intelligent Miner for Text). Other projects include word-sense disambiguation, linguistic support for IR, and relation extraction from text. Currently, she manages a question-answering project.

Roy J. Byrd *IBM Research Division, Thomas J. Watson Research Center, P.O. Box 704, Yorktown Heights, New York 10598 (electronic mail: roybyrd@us.ibm.com).* Mr. Byrd joined IBM in 1968 and has done development and research on programming languages and compilers, database query systems, object-oriented programming languages, natural language processing, information retrieval, and text analysis. From 1969 to 1974, he participated in the development of several IBM products incorporating compilers and interpreters for the BASIC and PL/I programming languages. From 1974 to 1976, he worked in the former IBM Advanced Systems Development Division on the EQUAL natural language query system for relational databases. He joined the T. J. Watson Research Center in 1976. Until 1981, he worked on the System for Business Automation project, which produced the Query-by-Example product, and which built a message-passing object-oriented programming system based on actor theory. Since 1982, he has done computational linguistics research on the analysis and construction of lexicons for natural language processing and on techniques for text analysis in large document collections. Mr. Byrd manages the Text Analysis and Language Engineering group in the Knowledge Management Technologies department at IBM Research. The group has contributed to several IBM products, including DB2® Text Extender and Intelligent Miner for Text. His current research interests include information extraction from large text corpora, using fast natural language processing techniques. He received his B.A. degree in theory and composition of music from Yale University in 1967 and is a Ph.D. candidate in linguistics at New York University.