Part III

Types of Information Retrieval Systems

In this section, we proceed to review many different types of information retrieval systems, with an emphasis on Web-based systems. Though some common internal mechanisms are used within IR systems, such systems also have to be adapted to the hugely varying circumstances of their use. One size does not fit all, and great improvements in performance can be attained through design for specific uses and audiences.

Randolph Hock, of Online Strategies, introduces "Search Engines" (Chapter 22), the key type of IR system used on the World Wide Web. Carlos Castillo and Ricardo Baeza-Yates, of Yahoo! Research, apply and adapt core concepts of information retrieval to the Web context in Chapter 23, "Web Retrieval and Mining." Kieron O'Hara and Dame Wendy Hall, a Fellow of the Royal Society, and one of the early developers of the Semantic Web, describe the "Semantic Web" in Chapter 24, of that name. The purpose of the Semantic Web is to provide structure and links among heterogeneous information sources to enable automatic computation and other forms of automatic processing on the underlying information. "XML Information Retrieval" (Chapter 25), described by Mounia Lalmas, of the Department of Computing Science, University of Glasgow, deals with the marking up of documents to enable targeted retrieval from portions of documents.

Yiyu Yao and colleagues, of the Beijing University of Technology, provide a model for IR systems that includes several kinds of support for the system and user, in order to improve overall performance, in Chapter 26 "Information Retrieval Support Systems." About these "IRSS" they say: "By moving beyond browsing, navigating, and retrieval, IRSS focus on a wide range of supporting functionalities, including summarization, exploration, analysis, knowledge discovery, results organization, retrieval strategy recommendation, and so on."

Each of the next three chapters addresses a major extension of conventional IR system design, making retrieval possible beyond the historical emphasis on monolingual text. Douglas W. Oard addresses "Multilingual Information Access" (Chapter 27); Vittorio Castelli, of IBM, lays out "Still Image Search and Retrieval" (Chapter 28); and Kjell Lemström and George Tzanetakis present "Music Information Retrieval" (Chapter 29).

Each of the remaining chapters in Section 3 addresses what could be called application areas—systems developed for particular environments. These areas, however, are in most cases so large themselves that they constitute their own research and development arenas. Whole conferences, professional associations, and grants programs exist to develop IR systems in these areas, and the ongoing

© 2012 Taylor & Francis Group, LLC

research continually improves the efficiency and effectiveness of information storage and retrieval, as well as our understanding of the human-system interface in a wide range of human situations.

The general importance of human contact and the explosive growth of online social media have generated their own research and development areas. Hady W. Lauw, of Microsoft Search Labs, and Ee-Peng Lim, of the School of Information Systems, Singapore Management University, explore "Web Social Mining" (Chapter 30) and Derek L. Hansen and colleagues drill in on "Recommender Systems and Expert Locators" (Chapter 31).

In the broad context of business and other organizational environments, Dick Stenmark presents "Knowledge Management Systems" (Chapter 32). He discusses the dimensions that are important in distinguishing and designing various types of knowledge management systems. A specific class of knowledge management systems, one at the heart of organizational performance, is covered in Chapter 33, "Decision Support Systems," as explained by Marek J. Druzdzel and Roger R. Flynn. Still another class of system supporting organizational performance, in this case by teams, is to be found in Chapter 34, "Collaborative Systems and Groupware," by David Jank.

Because of the value of geographical information to many business, government, and other institutions, geographic information has itself become a major research and development area. Timothy F. Leslie and Nigel M. Waters, of the George Mason University Department of Geography and Geoinformation Science, introduce the reader to this large area in "Geographic Information Systems (GIS)" (Chapter 35). Another area drawing vast sums of research and development money is medical information systems. In this context, Kai Zheng, of the University of Michigan Department of Health Management and Policy, introduces "Clinical Decision-Support Systems" (Chapter 36).

The section ends with several chapters on information systems in library, academic, and museum contexts. Libraries were, of course, the original information institutions, and, in the developing digital age, museums have joined libraries in needing to process and retrieve large amounts of digital information and images for their staff and for public users of museum Websites.

"Integrated Library Systems (ILS)," described in Chapter 37 by Emily Gallup Fayen, have been under development for over 40 years, helping libraries integrate the processing necessary to order materials, catalog them, provide an end-user catalog, manage circulation records, and de-accession resources, all in a single system reducing redundancy. The chief library information systems seen by library users, both in-house and remotely online, are the "Online Public Access Catalogs (OPACs)" (Chapter 38), described by Kevin Butterfield. These were one of the earliest information retrieval systems available to the general public, developed in the early 1980's, and they pioneered many IR innovations.

Public libraries have the added issue to deal with of ensuring that children not be exposed to unsuitable websites while searching the public computers online in libraries. Internet filtering is a kind of reverse IR system, identifying retrievals to *exclude*. Lynn Sutton, of the Reynolds Library of Wake Forest University in North Carolina, describes these systems and the social issues surrounding them in Chapter 39, "Internet Filtering Software and Its Effects."

For academics and students, the development of bibliographies and citation lists becomes a major information processing and retrieval issue when references grow into the hundreds or thousands. Dirk Schoonbaert and Victor Rosenberg review the history and state of the art of this class of small, but vital information retrieval systems in "Personal Bibliographic Systems (PBS)" (Chapter 40).

The section closes with three chapters on museums-related information systems. Museums need to keep IR systems of their sometimes vast collections, and each record within the system requires unusually extensive description—of the item itself, its provenance and legal standing, its history of curation, conservation, loans, etc. This special kind of database requires many kinds of retrieval capabilities and field types. Perian Sully, of the Judah L. Magnes Museum, and an expert in these "Collection Management Systems" (Chapter 41), describes them for the reader. Museums have been innovative in developing "Interactive Multimedia in Museums" (Chapter 42) for visitors. Nik Honeysett, of the J. Paul Getty Museum, describes this type of information system. Finally, David Bearman and Jennifer Trant, major originators and players in the world of museum informatics, describe the development and management of "Museum Web Sites and Digital Collections" (Chapter 43).

© 2012 Taylor & Francis Group, LLC

22 Search Engines

Randolph Hock

CONTENTS

© 2012 Taylor & Francis Group, LLC

INTRODUCTION

Web search engines, for the public at large, have come to be perhaps the most frequently used computer services for locating information. To some degree the same is true for many researchers, information professionals, and others. To most effectively and efficiently utilize these services, some understanding of the structure, make-up, content, features, and variety and breadth of these services is essential. This entry addresses those various aspects including just what is meant by "search engines," the components of a search engine, and typical search features, and it provides a profile of the major general Web search engines and a look at specialty search engines, visualization engines, and metasearch engines.

WHAT IS MEANT BY "SEARCH ENGINES"?

The term "search engines" can have a variety of meanings, in the broadest sense referring to any computer program that facilitates the searching of a database. In the context of library and information science, however, the term has come to primarily refer to "Web search engines," that is, those services on the Web that allow searching of a large database of Web pages and other Web content by word, phrase, and other criteria. (For this discussion, hereafter, "search engines" will be taken to refer to "Web search engines.") A certain level of ambiguity becomes apparent, however, when it is realized that what is often referred to as a "search engine" is often a reference to the overall service that is provided, beyond just a search of Web sites. ("Google" is thought of not as just the searching part of the Google enterprise, but the many added features and content as well.) It is often impossible and unproductive to discuss the narrower "searching" part without discussing the broader range of services. That ambiguity in terminology is a result and artifact of the history of search engines but recognition of the ambiguity is necessary for an understanding of the current nature of such services.

Search engines vary in a number of ways and most could be considered to fall into one of four categories: General Web Search Engines (which have the purpose of searching a large portion of all pages that exist on the Web), Specialty Search Engines (which focus on searching a specific kind of document, file type, or sources from a particular subject or geographic region), Visualization Search Engines (which furnish diagrams, images, or other "visuals" to show relationships among the items in a particular set of retrieved items), and Metasearch Engines (which gather together the search results on a specific topic from multiple search engines).

COMPONENTS OF A SEARCH ENGINE

General Web search engines and specialty search engines can be considered to have four major components that correspond to the steps required to create the service: 1) the identification and gathering of the material (Web pages, etc.) to be included in the engine's database; 2) an indexing program and the corresponding generated indexes; 3) the searching and ranking algorithms; and 4) the user interface.

Identifying Material to Be Included

Search engines identify those Web pages (and other items) to be included in the service's database by two means: "crawling" and submissions of pages. The first, "crawling" consists of having programs ("crawlers" or "spiders") that on an ongoing basis scan the Internet to identify new sites or sites that have changed, gather information from those sites, and feed that information to the search engine's indexing mechanism. The crawlers start by examining pages that the service already knows about and looking there for "new" links (links that the service does not already know about). When such links are identified, the pages to which the links led are likewise examined for "new" links, and

© 2012 Taylor & Francis Group, LLC

so on. More popular Web sites (such as those that have lots of links to them) may be crawled more thoroughly and more frequently than less popular sites.

The second way search engines identify new items to be added to the database is by having Web site owners (or others) "submit" sites or pages. Most engines provide a form by which this can be done. Search services maintain their own policies as to whether submitted (or for that matter, pages identified by crawling) will indeed be added to the database, particularly looking to exclude unacceptable content (spam, sexually explicit material, etc.)

Search Engine's Index and Indexing Program

After a new or changed page is identified by the search engine's crawler, the page will typically be indexed under virtually every word on the page (up to some usually undisclosed limit). In addition to text words, other parts or characteristics of the page may also be indexed, including the URL (Uniform Resource Locator, the "Web address"), parts of the URL, links, metadata found in the "head" of the document, the URLs of links on the page, image filenames, words in linked text, etc. By identifying and indexing these pieces of data (pieces or characteristics of the Web page or other type of indexed document, such as an Excel file), they become searchable "fields," thereby allowing users to use those fields to increase the quality of their search. The search system may also "derive" additional fields, such as language, by analysis of the document.

The Search Engine's Retrieval and Ranking Algorithms

By narrow definition, the actual search "engine" is the search service's retrieval program, that is, the program that identifies (retrieves) those pages in the database that match the criteria indicated by a user's query. That identification function is necessarily supplemented by another important and more challenging program that is used to determine the order in which the retrieved records should be displayed, based on measures that try to identify which retrieved records (pages, etc.) are likely to have the highest relevance in respect to the user's query.

This "relevance-ranking" algorithm usually takes many factors into account.

Exactly what factors go into the relevance ranking process varies, but they include: use of keywords in titles, text, headings, etc.; popularity of the sites (how many and which sites link to the site.); words used in anchors (clickable text); internal links (how many and what kind of links within the larger site point to the page); quality of links leading out to other pages (whether they point to high quality pages); etc.[1]

The success or the failure of the relevance ranking algorithm is critical to the user's perception of the search engine, the user's continued use of that system, and the commercial success of the engine.

The Interface Presented to the User for Gathering Queries

This interface the user typically sees includes the home page of the search service and other pages (such as an advanced search page) that present search options to the users and accept the users' search queries, as well as the search results page. The search service can choose to have their page focus almost exclusively on "search" (as with Google) or be a more general, wide-reaching "portal" page, providing much more than just searching capabilities. (The "portal" dilemma for search services will be discussed in more detail later.)

Regardless of what other services and information are provided on the service's homepage, the "searching" part usually consists of a single search box plus links to an advanced search page and to other searchable databases that are made available by the service (images, video, news, etc.) Usually there are also links to "help" screens, etc. While the simplicity of a single search box appeals to the less experienced user, it also usually provides substantial, but not obvious, capabilities for extensive searching sophistication, such as the potential for using Boolean logic and "prefixes" (e.g., "title:")

© 2012 Taylor & Francis Group, LLC

to perform field searching and other functions. The advanced search page much more explicitly lays out the possibilities to the user, providing a menu-based approach to utilization of features.

The Portal Dilemma

From the early days of search engines, search engine providers have wrestled with the decision as to whether to make their home page one that focuses almost exclusively on "search" or one that provides a variety of added services such as news, weather, etc., the latter approach often referred to as a "portal." From its beginning, before it was even a "search engine" and was just a directory, Yahoo! preferred the portal approach. AltaVista, a leading search engine in the 1990s, went back and forth between the two extremes, a situation which may have contributed to its demise. Google was, from the beginning, almost purely "search engine" and the simplicity of its interface was undoubtedly one factor in its rapid rise in popularity. Search services tend to "cover their bets" however, by providing alternatives. Yahoo! provides a Google-like option at search.yahoo.com and Google provides a personalizable Yahoo-style page with its iGoogle portal page.

Searching Options Typically Provided

All leading search engines provide a range of user accessible options that permit the user to modify their search queries in ways that can improve both the precision and the recall of their search results. Which specific options are provided varies from engine to engine, but there are several that are fairly typical (and some that are unique to a particular engine.) The most typical options include Boolean operations, phrase searching, language specification, and specifying that only those pages are retrieved for which the search term appears in a particular part (field) of the record such as the title, URL, or links. Since engines now cover other document types beyond just pages written in HyperText Markup Language (HTML), with several engines users can also narrow their search to a specific file format (Web pages, Adobe Acrobat files, Excel files, etc.). Most engines also provide an option to filter "adult content" material.

Boolean Logic

In the context of Web searching, "Boolean logic" refers to the process of identifying those items found in the database that contain a particular combination of search terms. It is used to indicate that a particular group of terms must all be present (the Boolean "AND"), that any of a particular group of terms is acceptable (the Boolean "OR"), or that if a particular term is present, the item is rejected (the Boolean "NOT").[2]

Engines usually provide two different ways to qualify a query with Boolean operations (1): the option of applying a syntax directly to what is entered in the search box and (2); menu options on an advanced search page. Using the menus can be thought of as "simplified Boolean" and, depending upon the structure of the advanced search page, may or may not provide the precision achievable by the use of syntax in the main search box (For example, the ability to apply "OR"s to more than one of the concepts included in the query may be done in the main search box but may not be allowed for on the advanced search page.)

The exact syntax used varies with the search engine. All major engines currently automatically apply an "AND" between your terms, so when the following is entered:

prague economics tourism

what will be retrieved is what more traditionally would have been expressed as: prague AND economics AND tourism.

Very precise search requirements can be expressed using combinations of the operators along with parentheses to indicate the order of operations. For example:

```
(grain OR corn OR wheat) (production OR harvest) oklahoma 1997
```

© 2012 Taylor & Francis Group, LLC

At various times, search engines have allowed the use of symbols (+, &, –, etc.) instead of words (AND, OR, NOT) and indeed, for the "NOT" most search engines currently suggest the use of a minus sign in front of the term. Some search engines require the use of parentheses around "nested" (OR'ed) terms, some do not.

For details on Boolean syntax for any search engine, the help pages for that engine should be consulted. There are also Web sites, such as Search Engine Showdown from Greg Notess (http://www.searchengineshowdown.com) that summarize the syntax (and other features) for all major engines.

The alternative to using syntax to apply Boolean is the use of menus on an advanced search page. There, for example, you may find a pull-down menu, where, if you choose the "all the words" option, you are requesting the Boolean AND. If you choose the "any of the words" option from such a menu, you are specifying an OR. There is usually also a box for excluding terms ("NOT").

Phrase Searching

Phrase searching is an option that is available in virtually every search engine, and almost always uses the same syntax, the use of quotation marks around the phrase. For example, searching on "Red River" (with the quotation marks) will assure that you get only those pages that contain the word "red" immediately in front of the word "river." Of all search engine techniques, this is widely regarded as one of the most useful and easiest for achieving higher precision in a Web search. It is also useful for such things as identifying quotations and identifying plagiarism.

Title Searching

Title searching, that is, limiting your retrieval to only those items (pages) that have a particular term or combination of terms in their title, is one example of "field searching," as referred to earlier. It is also another example of a technique that can yield very high precision in a search. Most search engines use the "intitle:" prefix and/or the "allintitle:" prefix for the syntax for title searching. ("allintitle:" allows specifying that more than one term be included in the title, not necessarily in any particular order.)

URL, Site, and Domain Searching

Search engines typically index Web pages (and other document types) by both the overall URL and by the segments of the URL. This facilitates the finding of any document that comes from a particular domain or part of a domain (also a specific site or part of a site). Doing a search in which results are limited to a specific site allows one, in effect, to perform a search of that site. Even for sites that have a "site search" box on their home page, more complete results can often be found by using this technique than by using the site's own search feature. "inurl:", "allinurl:," and "site:" are the prefixes commonly used.

The term, "Domain searching" is sometimes used to refer to the above process and the use of the term, "Domain," points out that this approach can be used to limit retrieval to sites having a particular top-level domain, such as: gov, edu, uk, ca, or fr. This could be used, for example, to identify only Canadian sites that mention tariffs, or to only get educational sites that mention biodiversity.

Link Searching

There are two varieties of "link" searching. In the more common variety, one can search for all pages that have a hypertext link to a particular URL, and in the other variety, one can search for words contained in the linked text on the page. In the former, you can check, for example, which Web pages have linked to your organization's URL. In the second variety, you can see which Web pages have the name of your organization as linked text. Either variety can be very informative in terms of who is interested in either your organization or your Web site. Also, if you are looking for information on an organization, it can sometimes be useful to know who is linking to that organization's site.

© 2012 Taylor & Francis Group, LLC

This searching option is available in some search engines on their advanced page and/or on the main page with the use of prefixes. (usually "link:"). Engines may allow you to find links to an overall site, or to a specific page within a site.

Language Searching

Although all of the major engines allow limiting retrieval to pages written in a given language, they differ in terms of which languages can be specified. The 40 or so most common languages are specifiable in most of the major engines. Though some engines provide a prefix option for searching for languages, more typically one would go to the engine's advanced search page to narrow to a language.

Date Searching

Searching by the date of Web pages is an obviously desirable option, and most major engines provide such an option. Unfortunately, because of lack of clear or reliable information on a page regarding when the page itself was initially created, the date on which the content of the page was created, or even when the content on the page was significantly modified, it is often impossible for a search engine assign a truly "reliable" date to a Web page. As a "workaround," engines may take the date when the page was last modified or may assign a date based on when the page was last crawled by the engine. For searching Web pages, users should be aware of this approximation and its effect on precision when using the date searching option that is offered by most search engines (usually on their advanced search page). (On the other hand, for some of the other databases an engine may provide, such as news, the date searching may be very precise.)

Searching by File Type

For most of the 1990s, most search engines only indexed and allowed searching of regular HTML pages. In the crawling process (or for submitted pages) when the engine's indexing program encountered a link that led to another type of document, such as an Adobe Acrobat (pdf), or Excel (xls) file, the link was ignored. Starting with Adobe Acrobat files, other file types were fairly rapidly added to the corpus of "indexable" pages. This not only increased the breadth of resources available to the searcher, but also provided the capability for the searcher to limit retrieval by type of file. Limiting to Adobe Acrobat files provides documents more suited to printing. Narrowing to PowerPoint files can provide convenient summaries of a topic. Limiting to Excel files can often enable a greater focus on statistics.

Search Results Pages

As well as providing enhanced searching capabilities, search engines also enhance the content of results pages, beyond presenting just a listing of the Web page results that match the user's query. At the same time they search their Web database, they may automatically search the other databases they have, such as news, images, and video, and on search pages may automatically provide links to the matching items from those additional databases. Some search engines may search additional "reference" resources, such as dictionaries, encyclopedias, maps, etc., and likewise display matching content from those sources.

As well as displaying such supplemental content on results pages, search engines may also provide suggestions for ways in which the user might further qualify search criteria. This is done by suggesting related, narrower, or broader topics. Some engines also provide links to narrow the search by file type, language, or type of site (weblog, forum, commercial or noncommercial, etc.)

Specific options may also be offered on results pages for each retrieved item. Some engines keep a copy of each page they have indexed and provide a link to that "cached" page. This is particularly useful if, in the time since the page was indexed, the page was removed, is not available because of a server problem, or has changed in a way such that the term the user searched for is no longer on the page.

© 2012 Taylor & Francis Group, LLC

With records for pages that are not in the language of the search engine interface, there may be an option to translate the record (for example, if the user is using an English language version of Google and a page is in French or if the user is using the French version and the page is in English). Click on the "translate" link to receive a machine translation of the page. As with other machine translations, what you get may not be a "good" translation, but it may be an "adequate" translation, adequate in that it will give you a good idea of what the page is talking about. Also keep in mind that only "words" are translated. The translation program cannot translate words you see on a page that are actually "images" rather than "text."

One feature offered on search results pages by all of the major engines is a spell-checker. If you misspelled a word, or the search engine thinks you might have, it graciously asks something like "Did you mean?" and gives you a likely alternative. If it was indeed a mistake, just click on the suggested alternative to correct the problem.

Search results pages will usually display links labeled as "Sponsor Results," "Sponsored Links," etc.—These are "ads" for Web sites and are there because the Web site has paid to appear on the search engine's results pages. Major engines keep these sponsor sites clearly identifiable by, for example, putting them in a blue background, or to the side of the page. Searchers should remain aware that it is the presence of these ads that makes the existence of search engines possible.

The Search Engine Leaders—Post-2000

Popularity of various search engines can change fairly quickly. In the early and mid-1990s a list of the most popular engines included, among others, AltaVista, Hotbot. Excite, InfoSeek, and Lycos, (Yahoo! was still a primarily a directory, and though it had a search engine function, for that function it made use of, at various times, AltaVista's and Google's databases.)

By the latter part of the 2000s the following were the leaders: Google, Yahoo!, MSN/Windows/Live Search, AOL, and Ask. (in that order). Those five search engines represented 94% of all (U.S.) searches.[3] (Brief profiles of the engines just mentioned are given below.)

Google

Google, which emerged as a company in 1998, grew very rapidly, its growth attributed largely to the simplicity of its interface, the lack of advertisements on the home page, and the quality of its relevance ranking (that fact significantly affected by Google's patented PageRank program.)[4] Google rather quickly went beyond "search" and began providing additional features and content, some of the enhancements emerging from within the Google organization and some (such as its e-mail service, Gmail) being patterned after such services already offered by its competitors. By the late 2000s, Google claimed more of the search market than all of its competitors combined and was offering a broad range of search services and a number of services not directly related to search

For its Web search offerings, Google provides all of the typical search options (Boolean, field searching, etc.) plus some unique searching features, the latter including numeric range searching (e.g., china history 1850 . . . 1890), and synonym searching (e.g., ~cars). As well as the searching of Web pages, Google also offers searches of databases of images, maps, news, products, video, groups, books (Google Book Search), journal articles (Google Scholar), and blogs. Some of these search offerings are very similar to corresponding services offered by Google's competitors, but some, such as "Google Book Search," were original and regarded by many as "ground-breaking" and even in some cases, controversial. (Google Books Search is a major book digitalization project, in cooperation with major publishers and libraries.) The search features provided with each of these databases is typically tailored to the specific nature of that kind of content.

Many of Google's Web search features are features that were already found on other search engines, but for which Google provided significant enhancements. One example is Google Language Tools. Many search engines have provided a translation option that allows retrieved items from a

© 2012 Taylor & Francis Group, LLC

number of non-English languages to be translated, using programs such as SYSTRAN's Babel Fish. In 2007, Google enhanced its own translation feature by allowing the user not just to translate a specific result, but to input a search in the user's own language, then have Google automatically translate the search terms, perform the search, and then deliver results in both languages. Translations are done using Google's own statistical translation technology.

As it grew, Google rather rapidly redefined itself to be much more than a "search engine," adding services that went beyond "search" and even beyond usual Web site content. Some services had a direct relationship to "search," such as Google News Alerts, Google's financial portal ("Google Finance"), the Google Toolbar for Web browsers, a desktop search tool for searching the content of one's own computer, and Google's own Web browser ("Chrome"). Some of the services Google began to offer included types of things that already existed as "portal" features in other search services. These offerings included a customizable portal page (iGoogle) with Google's own calendar and notebook and links to a variety of other content such as newsfeeds. Among other services are Gmail (a Web-based e-mail service), Google Earth (imagery and related geospatial content for the entire Earth, as well as the Moon and the sky) and Google Talk (an instant messaging service). One of the manifestations of "Web 2.0" is availability of user-accessible software that is resident on the Web, rather than on the user's own computer. (The term, "Web 2.0," refers not to an actual "version" of the Web, but to the fact that the nature of the Web, by the middle of the first decade of the twentieth century, had changed from being primarily a place to go to find information to being a place that was much more personal, interactive, and collaborative, with the Web as a "platform" where programs are provided, used, and shared.) Google has moved very much in the Web 2.0 direction, providing Picasa (a photo-sharing and editing service), SketchUp (a computer-aided design, CAD, program), Google Docs (a collaborative spreadsheet, word-processor, and presentations program), and Sites (for creating Web sites). Google also offers "mobile" services (including mobile search, maps, text messaging, Gmail, etc.), an enterprise version of Google's search engine, and a custom search engine that allows a user to have a search box (on their own Web site or as a page on Google) that delivers a search of only the user's own selection of Web sites.

Yahoo!

Yahoo! was among the earliest Web sites that had the purpose of leading users to specific content on the Web. In the beginning, Yahoo! was exclusively a "Web directory," a categorized list of selected Web sites. By 2000, however, it had begun a transformation to a portal site, having, in addition to the directory, over three dozen links to news, services, and other resources provided by Yahoo and its affiliates, including pages for shopping, auctions, phone numbers, a calendar, and more. From its earliest days, the Yahoo! homepage contained a search box, but results for that search came from a search of the directory, and later a search of Web databases from other search providers.

Yahoo!'s directory function became less and less central and in 2004 Yahoo! created its own database of Web pages. Though emphasis on "search" continued to increase and the emphasis on the directory declined significantly, Yahoo!'s main image continued to be that of a portal, with the emphasis on the wide range of other services provided by Yahoo! and its partners, including Yahoo!'s highly popular e-mail service and its sections on autos, finance, games, groups, health, job listings, maps, real estate, travel, and over 50 other content areas.

In the area of Web search, Yahoo! currently provides typical Web search features such as Boolean and field searching, though a continued absence of a link on its main page to its advanced search page reinforces the impression of Yahoo!'s preference for a portal focus over search focus. It's personalized portal page, My Yahoo!, is judged by some to be the most popular portal on the Web.[5]

In addition to Web search, Yahoo! offers searching of the following databases: news, images, video, maps, local (businesses), shopping, audio, jobs, Creative Commons, people (phone numbers and addresses), and travel reservations search.

© 2012 Taylor & Francis Group, LLC

MSN/Live Search

Microsoft has made several attempts since the mid-1990s to produce a Web search engine that is competitive with Google and Yahoo!. The attempts, made available primarily through Microsoft's MSN portal, have gone by a variety of names, including Microsoft Search, MSN Search, Windows Live, and, in 2008, Live Search (live.com). Search features have varied considerably and have at times been less robust than those of its competitors. Live Search presented some innovative features such as a design that allowed continuous scrolling through search results, but it, like some other features in the MSN search products, was short-lived. The 2008 version provided the typical Boolean and field searching options, plus some additional options such as "prefer:" by which the user can adjust the ranking weight for search terms, and "feed:" and "hasfeed;" which identify Web sites that contain RSS links on the user's chosen topic. In addition to the search for Web pages, Live Search also offers searches for images, video, news, maps, health information, local (businesses), products, and travel.

AOL

AOL Search is the search engine found on AOL's main portal page and is also available at search.aol.com. The search is provided in conjunction with Google and Web search results come from the Google database (but are typically fewer in number than when the search is done on Google itself. AOL Search also provides options for searching images (using Google), video ("Powered by TRUVEO"), news, shopping, jobs, maps, movies, music, personals, travel, and yellow pages.

Ask

Ask, which was formerly AskJeeves, underwent a number of significant changes as it changed from the "question and answer" format of the original AskJeeves. Ask created a substantial Web database with fairly typical search functionality, though missing some features such as an OR Boolean function. In 2008, the company underwent a reorganization which produced some doubts among those who watch search engines as to Ask's commitment to "search." As well as its Web search, with Ask you can also search databases of images, news, maps, businesses, shopping, TV listings, events, videos, recipes, and blogs. Results pages for Web searches automatically incorporate results from multiple databases and provide a "binoculars" icon for previewing results without leaving the results page.

Other General Search Engines

There are a number of other general Web search engines, including GigaBlast, Exalead, and others. Exalead (http://www.exalead.com/search), from France, incorporates a number of features unavailable in other current search engines, including truncation ("words starting with"), phonetic spelling, approximate spelling, and NEAR. These are important to note because they are reflective of a level of sophistication of search techniques a bit closer to those found in commercial search services such as Lexis/Nexis, Factiva, and DIALOG, but not found in Web search engines.

Specialty Search Engines

Over the years, a variety of search engines have appeared that could be classified as "specialty" search engines. Among these there have been attempts to create search engines that focus on a particular topic or geographic location. In most cases, an examination of these showed that what was provided was more of a "directory" of selected sites than a broad ranging crawler-based search of Web pages for the specific topic or locality. On the other hand, there have been many successful attempts to produce search engines that provide searching for a particular format or type of document, such as images, video, blogs, forums, etc.

© 2012 Taylor & Francis Group, LLC

News

Searching of news databases is available from all of the general Web search engines. There are numerous other Web sites that specialize in searching news content. Each of these have varying degrees of searchability, and from the research perspective it is important to note that the coverage can vary significantly, especially in regard to the number of news sources included, the time span for the content of the database, and the languages covered. Among the better-known news search engines are: NewsNow, Silobreaker, NewsExplorer, RocketNews, Topix.net, World Press Review Online, and NewsTin.

Images

The most commonly encountered image search engines are those that are included as databases provided by the general Web search engines, including Google, Yahoo!, Live Search, AOL, and Ask. As well as subject searching, most of these engines allow for Boolean, and narrowing by size, coloration, site, and adult-content filtering. On Google's advanced image search page you can also narrow to news or photo content, or those that appear to include faces. Flickr (flicker.com), an image sharing Web site, has also gained extensive popularity as an image search engine. The extensive tagging of photos by Flickr users makes millions of images searchable. PicSearch provides an extensive collection of images from the Web and in addition to the above search criteria also allows narrowing to animated images. There are also image search engines such as Corbis, Fotosearch, and Stock.XCHNG which enable users (for a fee) to have use of photos from commercial photographers and photo archives.

Video

As with image searching, searching for video is available from major search engines, including Google, Yahoo!, Live Search, and AOL. Extensive searching of videos produced by individuals, as well as commercial video, is available from YouTube, the leading video-sharing site. Depending upon the search engine, options are provided for searching by Boolean, language, duration, domain/site/source, format, popularity, aspect ratio, and resolution, plus filtering for adult-content. Some video search engines specialize in video from TV, including news programs, interviews, etc. These include Blinkx (free) and TVEyes (fee-based). Both of these utilize voice-recognition technologies to create searchable transcripts for their video content.

Forums

Content found in forums (discussion groups, groups, newsgroups, etc.) can be utilized for a number of applications, ranging from hobbies to tracking terrorist activities, and there are search engines that specialize in finding this category of document. Among the search engines that provide such access to forums from multiple sources across the Web are BoardReader and OMGILI. (There are a number of other places where groups can be searched, such as Google, Yahoo!, Topica, Delphi Forums, but those sites focus on searching only the content that is hosted on their own Web sites.)

Other Specialty Search Engines

There are still other categories of specialty search engines, including those for searching blogs and RSS feeds (examples: Technorati, IceRocket, Bloglines, and Google Blog Search), for searching podcasts (examples: Podcastdirectory.com, Podcast.com), and for searching for information on people (examples: pipl Search, Infobel, Yahoo People Search, Intelius, PeopleFinders).

Visualization Engines

Visualization search engines are Web sites (or programs) that provide a very different "look" (literally) at search results. Instead of the traditional linear, textual list of retrieved items, results are

© 2012 Taylor & Francis Group, LLC

shown on a map that spatially shows conceptual connections. Most current visualization engines utilize not a database of their own, but borrow one from other engines (Google, Yahoo!) or other sites such as Amazon.com. Visualization has been, and continues to be, an area of extensive research and there are several sites that demonstrate various visualization approaches. The type of conceptual and visual mapping done by these sites can be especially useful for quickly exploring the concept possibilities, directions, and terminology for a particular search. It presents a "connect the dots" approach, enabling understanding relationships among the concepts found in various search results—rather than just browsing lists of results. Among the leaders in this area are Kartoo, TouchGraph, and Grokker, and Quintura.

Metasearch Engines

The term "metasearch engine" (or "metasearch site") usually refers to Web sites that search multiple search engines in a single search. The degree of overlap (or lack thereof) between search engine results is something that professional searchers frequently consider and allow for as they search and searching more than one engine is a widely encouraged technique. Metasearch engines have been available since the 1990s and include sites such as Dogpile, Clusty, Ixquick, Mamma. Search.com, and many others. Each of these may provide additional benefits beyond just a compilation of results from more than one engine, for example, the "clustering" (categorization) of retrieved results, a feature that may not be provided by the target engines themselves. However, users should be aware of several shortcomings that may be encountered with these tools: 1) most of the current metasearch engines do not cover the largest major engines, particularly Google and Yahoo!, which tend to block queries from metasearch engines; 2) metasearch engines typically only return the first 10–20 results from any of the "target" engines; 3) metasearch engine results often discard useful and search-relevant information found on the actual search engine's results pages; 4) metasearch sites, even if they do cover the largest engines, may be required by those engines to show paid listings first; and 5) metasearch engines typically do not allow application of many of the search features available in the target engines themselves.

Metasearch engines should be distinguished from "comparison search" sites, such as Zuula.com and Twingine (twingine.no) which provide more of a side-by-side comparison of actual results from the target engines.

CONCLUSION

Web search engines have evolved significantly since they were first introduced in the early 1990s. The basic concept has remained the same, but the quality of results, the size of their databases, and the types of material that they include have increased dramatically. The total number of general Web search engines "in the race" has decreased and at present is dominated by one service, Google. Where the field of players has expanded is in the area of specialty search engines which focus on a specific type of Web "document." What has evolved even more dramatically is the "mission" of search services, which particularly in the case of Google, has gone far beyond "search." With advancing technologies, increasing interactiveness of the Web, and a more and more Internet-centered society, users can expect continued, fast-paced innovation.

REFERENCES

1. Sullivan, D. Ranking the SEO ranking factors. *Search Engine Land*; searchengineland.com/ranking-the-seo-ranking-factors-10890.php (accessed April 2009).
2. Bednarek, A.R. Boolean Algebras [ELIS Classic]. *Encyclopedia of Library and Information Sciences*, 3rd Ed.; Taylor & Francis: New York, 2009; 660-665..
3. Nielsen//Netratings Announces August U.S. Search Share Rankings. New York, September 19, 2007. NetRatings, Inc. http://www.nielsen-netratings.com/pr/pr_070919.pdf (accessed April 2009).

© 2012 Taylor & Francis Group, LLC

4. Vise, D.; Malseed, M. *The Google Story*; Bantam Dell: New York, 2005; 37–40.
5. About.com. Web Trends: The Top Ten Most Popular Portals on the Web. February 25, 2008. http://webtrends.about.com/od/webportals/a/topten_portals.htm (accessed April 2009).

BIBLIOGRAPHY

1. Search Engine History. http://www.searchenginehistory.com.
2. Search Engine Showdown. http://www.searchengineshow-down.com.
3. Vise, D.; Malseed, M. *The Google Story*; Bantam Dell: New York, 2005.

© 2012 Taylor & Francis Group, LLC

23 Web Retrieval and Mining

Carlos Castillo and Ricardo Baeza-Yates

CONTENTS

INTRODUCTION

Information retrieval is the area of computer science concerned with the representation, storage, organization, and access to documents.

Documents, in this definition, are understood in a broad sense, and include Web pages and other contents available on the Web. The Web is an unique medium for information dissemination, characterized by low entry barriers, low publishing costs, high communication speeds, and a vast distribution network.

Most methods for information retrieval were developed in the 1970s and 1980s for relatively small and coherent collections, such as the ones found in traditional libraries. The Web poses significant challenges to these methods, being massive, dynamic, and distributed.[1]

Web information retrieval (Web IR) or Web search, differs significantly from traditional information retrieval. The two main differences are the scale and nature of the collections being processed. Web search includes topics such as Web crawling, indexing and querying, adversarial Web IR issues, and Web distributed systems and evaluation metrics. Another relevant topic is Web data mining, which includes the analysis of the content, structure, and usage of the Web. In the following, we focus on these two topics, Web search and Web data mining. Our coverage of details and bibliography is by no means complete, and the interested reader is referred to Baeza-Yates and Ribeiro-Neto[2] and Chakrabarti.[3]

WEB SEARCH

Web search is the main application of Web IR, and a very successful one. From the user's point of view, a short query consisting of a few keywords is written in a search box, and the search engine

© 2012 Taylor & Francis Group, LLC

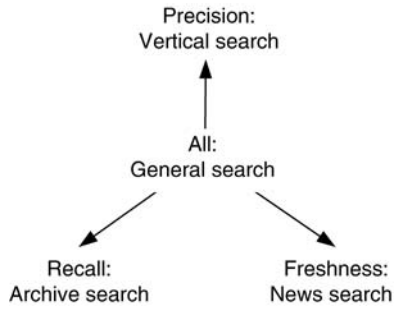

FIGURE 23.1 Trade-offs of different search engines.

displays in return a short list, typically of 10–20 Web pages that are considered relevant to the query issued and expected to be high-quality documents.

The two main goals in search are precision and recall, and they are, to a certain extent, competing goals.

Precision is defined as the fraction of relevant results contained in the result set, or in a part of the result set. For instance, if 3 out of 10 results for a query are relevant, the precision is 30%.

Recall is defined as the fraction of relevant results in a set, compared with the total number of pages on the Web that would be relevant for this query. Of course, the total number of pages on the Web relevant for a particular query is an unknown quantity, but for popular query terms it can be estimated using sampling techniques.

An information retrieval system can have high recall at the expense of precision, simply by returning more results, and high precision at the expense of recall, by removing results for which the algorithm is unsure about their relevance. The design of effective algorithms for search seeks a balance among these two extremes, and in the Web the focus is on precision as recall cannot be measured, only estimated.

In the case of Web search there is a third goal that is freshness. The Web changes continuously and the copy of the Web that the search engine has can become stale very quickly. The three goals: precision, recall, and freshness are sometimes mutually exclusive and introduce three-way trade-offs,[4] as depicted in Figure 23.1. These trade-offs create the possibility of several niche markets apart from general Web search, including: vertical search, over a particular subset of pages; archive search, over several snapshots of the Web; and news search, over Web sites that change with very high frequency.

An additional consideration in search engine design is efficiency. Large Web search engines have to deal with a large volume of queries and search huge data collections, so even large amounts of computational resources can be insufficient. Successful algorithms for Web search avoid consuming too many resources per query or per document.

From the point of view of the search engine, Web search occurs in two main phases. The first phase is off-line, with a certain periodicity or by permanent incremental maintenance. It includes crawling the Web to download pages and then indexing them to provide fast searches. The second phase is done online, and corresponds to the process of querying and ranking, which consists in building a ranked list of results using the index for a particular query. These phases are depicted in Figure 23.2 and explained in more detail in the rest of this section.

Web Crawling

A Web crawler is a system that automatically downloads pages from the Web following a set of predefined rules. A Web crawler receives as input a starting set of URLs that constitutes a "seed set," and a set of rules to follow. The crawler first downloads the pages from the seed set, extracts the links found in such pages, and then follows those links recursively while certain criteria are met. Crawling the Web is a required step for many Web IR applications.

© 2012 Taylor & Francis Group, LLC

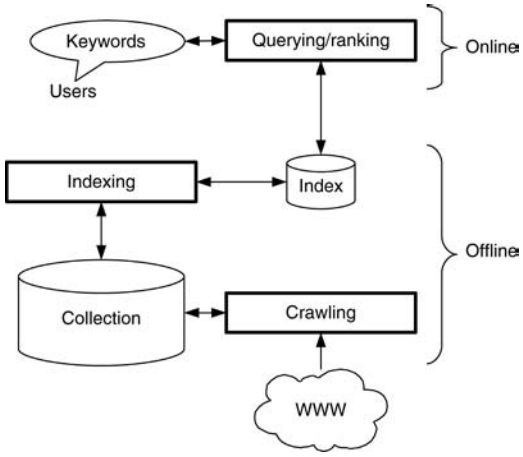

FIGURE 23.2 Phases of Web search.

Aside from Web search, Web crawlers are multipurpose systems that can be used for a variety of tasks, including finding and reporting "broken links" or other coding errors, and computing statistics about the Web.

The most important design constraint of Web crawlers is that they must avoid disrupting the Web servers they interact with. While downloading Web pages, the crawler is using the resources of others, and thus it must keep its resource consumption as low as possible. Web crawler designers and operators must take every possible step to control the frequency of visits to sites and keep them to a minimum. Also, the authors of Web sites have to ultimately decide which part, if any, of their sites can be visited by crawlers. This is done by using the robots exclusion protocol.[5]

After downloading the pages, they have to be processed to be used by the search engine or other application. HTML is the main language for coding documents on the Web, but there are many other formats present, including PDF, plain text, plus the document formats used by popular text-processing software such as Microsoft Word or OpenOffice. These formats have to be converted to a single representation before they can be used.

The importance of freshness is another aspect of the crawler's operation. The Web is very dynamic, and it changes continuously; this means that by the time the crawler has finished collecting a set of pages, many of the pages it has downloaded have already changed.[6] Crawling the Web, to a certain extent, resembles watching the sky at night[2]: the light we see from the stars has often taken thousands of years to reach our eyes. Moreover, the light from different stars has taken different amounts of time, so what we see is not a snapshot of the sky at any given moment, present or past. It is a combination of images from different times. The same happens with the collection of Web pages crawled by a search engine.

The Web pages that are not directly accessible by following links, but require the user to enter a query in an online form (e.g., enter an author's name to retrieve bibliographic data), constitute the "hidden Web."[7] Searching this content is challenging for search engines. In most cases, large information providers generate "crawler-friendly" pages for better indexing by search engines, but other forms of collaboration may arise in the future, including exposing an interface for querying the local database to the search engine.

Indexing

After collecting pages, the next step is to create an index to enable fast searches over the downloaded pages. The first step toward indexing a large collection is to consider an appropriate logical view of the content. The most used logical view for this task is the "bag of words" model,[8] in which

© 2012 Taylor & Francis Group, LLC

each document is represented as a multiset containing all its keywords, disregarding the order in which they appear.

To produce this logical view, text normalization operations are applied to the original texts. These operations include tokenization, stopword removal, and stemming.

Tokenization is the process by which a text is separated into words. This is trivial in Western languages, but harder to do in other languages such as Chinese.

Stopwords are functional words that do not convey meaning by themselves, such as articles and prepositions. The removal of stopwords reduces the amount of data processing and the size of the index, and also improves the retrieval accuracy of information retrieval systems.

Stemming is the extraction of the morphological root of a word. This allows us to search for "housing" and retrieve results that include "house" or "houses."

After the text normalization operations have been applied, most search systems build an index, a data structure designing to accelerate the process of retrieving documents containing a given query. The most prevalent type of such structure is an inverted index. In Figure 23.3, an example of an inverted index for a collection of five toy documents (each of them having two words) is shown.

An inverted index is composed of two parts: a vocabulary, containing all the terms in the collection, and a posting list, which contains references to the documents(s) in which each word of the vocabulary appears. An inverted index is a powerful tool for the search engine, enabling very fast response times. In the example of Figure 23.3, if we search for "global AND climate" in the inverted index, the task is basically to intersect the set of pages containing "global" {1, 2, 5} with the set of pages containing "climate" {2, 3, 4}, obtaining as a result the set {2}. If these lists are sorted, their intersection can be computed very quickly. This is how a basic inverted index works. There are many techniques for providing faster search or reducing the space occupied by the index. For example, if phrase or proximity search is needed, the exact positions where the term appears in a document must be also encoded in the posting list. The interested reader is referred to Baeza-Yates and Ribeiro-Neto[2] and Witten, Moffat, and Bell[9] for an overview of indexing techniques.

Another aspect is that large search engines achieve high response times by means of parallelization. In this case, the index has to be divided in some way, and each piece of the index has to be given to a different physical computer. There are two main strategies for this partitioning. One is to give each machine a set of documents, the other one is to give each machine a set of terms.[10]

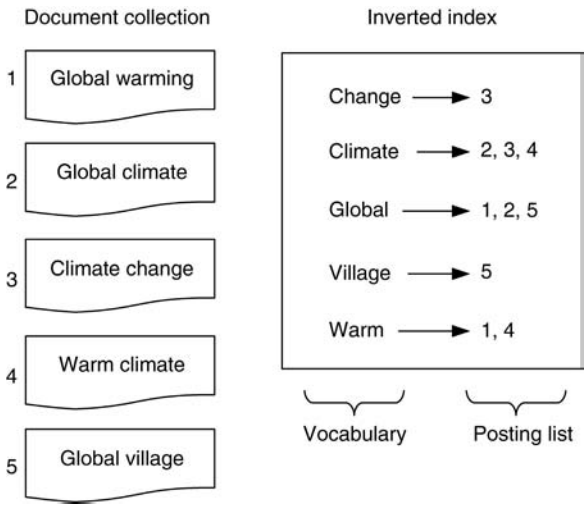

FIGURE 23.3 Example of an inverted index.

© 2012 Taylor & Francis Group, LLC

Querying and Ranking

Most search engines receive queries expressed as a set of keywords. Scalable question answering systems, in which users express their information need by means of a question, have remained elusive to researchers in particular because many natural language processing algorithms still require a prohibitive amount of computational power for Web-scale collections.

Typical queries are very short, between two and three keywords each. After receiving a query, the search engine uses its inverted index (or indexes) to build a page with results that is shown to users. To a certain extent, the problem of finding a set of pages that are related to the query is the "easy" part, given that for most broad queries there are thousands or millions of documents that are potentially appropriate. The most difficult challenge is to find among those documents, a small subset of the best 10 or 20. This is the problem of ranking.

Ranking has two main aspects: relevance and quality. The dimension of relevance indicates how related is the retrieved document to the user intention. The dimension of quality indicates how good is the document by itself. Search engines try to produce results for a given query that are both relevant for the query and have high quality. One of the main techniques to do fast ranking is to use partial evaluation techniques, such that only the top ranked answers are computed, and the rest of the answer is computed incrementally as the user demands it.

Relevance

Given that the search engine cannot understand the meaning of the queries nor of the documents, it must resort to statistical methods to compare queries to documents. These statistical methods allow the search engine to provide an estimation on how similar the query is to each document retrieved, which is used as an approximation of how relevant is the document for the query.

The vector space model[8] is the most used framework for measuring text similarity. It represents each document as a vector in a high-dimensional space, in which each dimension is a term, and the magnitude of each component of the vector is proportional to the frequency of the corresponding term, and inversely proportional to the document frequency of the term in the collection.

Differences in document size have to be taken into account for the similarity measure between documents, so the angle between documents is used instead of, for instance, the Euclidean distance between them. For instance, the angle between the documents "global warming" and "warming warming global global" is zero (so the documents are equivalent according to this metric), the angle between the documents "global warming" and "global climate" is 45° (under a simple weighting scheme), and the angle between the documents "global warming" and "climate change" is 90°. For normalization purposes, the cosine of such angle is the standard way of expressing this similarity metric.

Information retrieval systems usually do not apply the vector space model naïvely, as it has significant weaknesses. By itself the vector space model does not take relationships among terms into account.[11] For instance, strictly speaking the cosine similarity between the "global warming" and "climate change" is zero, and the cosine similarity between "global warming" and "strawberry ice cream" is also zero; but clearly the first pair of concepts have a closer relationship than the second pair. Two methods that can be applied to overcome this problem are query expansion and latent semantic indexing.

Query expansion consists in adding related words to the queries, and the same technique can be applied to documents. For instance, this could convert automatically "global warming" into "global world warming climate" and "climate change" into "climate warm cold change global." The specific words that are added can be obtained from different sources, including cooccurrence in the collection. In the case of the Web, there are rich sources of information to obtain words related to a document. The main one is anchor text, that is, the text contained in the links pointing to the current document. This is a very important feature in the ranking computed by most modern search

© 2012 Taylor & Francis Group, LLC

engines. A second source of information are social book-marking sites that allow users to associate tags to documents.

Latent semantic indexing[12] consists in projecting the vectors representing queries (and documents) into a different, and usually smaller, space. This technique is based on principal component analysis and attempts to group automatically terms into the main "concepts" representing multiple weighted terms.

Quality

Search engines are designed to extract a set of features from the documents they index, and use those features to assert what is the quality of a given document. Quality is hard to define and of course hard to estimate using statistical measures. However, certain textual features from documents, including content length, frequencies of some words, features about the paragraphs, etc. tend to be correlated with human assessments about document quality.[13]

Apart from the content of the pages themselves, on the Web a rich source of information for inferring quality can be extracted from links. Links on the Web tend to connect topically related pages,[14] and they often imply that the target document has an acceptable or high level of quality. Thus, they can be used for finding high-quality items in the same way as academic citations can partially characterize the importance of a paper. The same considerations as for academic citations apply: not all of the links imply endorsement,[15] some pages attract many citations for other reasons aside from quality, and citation counts can be inflated by self-citations or citations that point to errors; among other problems.

There are two classic link analysis algorithms to obtain quality metrics for Web pages: PageRank and HITS. For a survey of their variants, and other methods, see Borodin, et al.[16]

The PageRank algorithm[17] defines the importance of a page in a recursive manner: "a page with high PageRank is a page referenced by many pages with high PageRank." Despite the definition being recursive, it is possible to compute PageRank scores using results from Markov chain theory. In brief, the wanderings of a "random surfer" are simulated, in which a person browses the Web by following links at random. The PageRank score of a page is roughly proportional to the amount of expected visits the random surfer will do to each page.

The HITS algorithm[18] is another method for ranking Web pages. It starts by building a set of pages related to a topic by querying a search engine, and then expands this set by using incoming and outgoing links, by crawling the Web or by querying a search engine again. Next, two scores for each page are computed: a *hub score* and an *authority score*.

As shown in Figure 23.4, a page with a high hub score is a page that links to many pages with a high authority score. A page with a high authority score is a page linked by many pages with high hub score. Again, despite the apparent circularity of the definition, both hub and authority scores can be computed efficiently by an iterative computation.

Another source of information for ranking pages on the Web is usage data. A page that is visited frequently and/or for long periods by users may be more interesting than a page that is not. This information can be obtained by the search engine by providing a client-side add-on such as a toolbar, or by instrumenting the search engine result pages to capture click information.

Ranking Manipulation

Visits from search engines are an important source of traffic for many Web sites. Given that in the case of commercial ventures on the Web, traffic is strongly correlated with sales volume, there is a significant economic incentive for obtaining high rankings on search engines. These incentives may lead Web page authors to use deceptive techniques for achieving high rankings.[19] These deceptive techniques are known as a whole as search engine spam.

There are many types of search engine spam: inserting many keywords on Web pages, linking nepotistically among pages, providing different contents to the search engine than to users (also called "cloaking"), among others; for a survey of these methods, see Gyöngyi.[20]

© 2012 Taylor & Francis Group, LLC

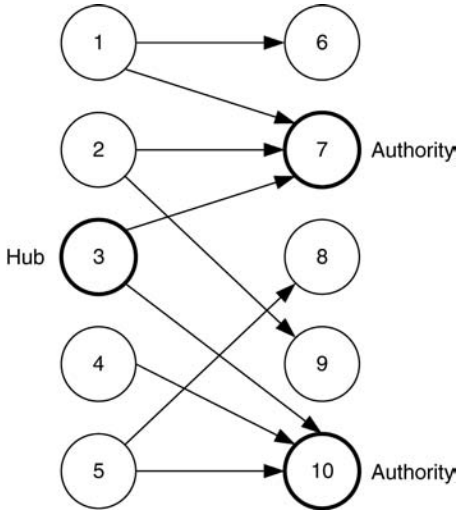

FIGURE 23.4 A graph with one page with high hub score (number 3) and two pages with large authority scores (numbers 7 and 10).

Search engine spam has been an important issue for search engines for a number of years, and it is not likely to be solved in the near future. Web spam damages search engines' reputation as it exploits and weakens the trust relationship between users and search engines.[20] Spamming has become so prevalent that without countermeasures to identify and remove spam, the quality of search engine results would be very poor.[21]

WEB MINING

Web mining is the application of data mining techniques to find patterns on data downloaded from the Web. Based on the main source of data they use, these techniques can be broadly classified as Web content mining, Web link mining, and Web usage mining.

Content Mining

Web content mining is the extraction of knowledge from the textual content of Web pages. The main challenge here is that HTML, while designed initially to be a language for logical formatting, is actually used as a language for physical formatting. Logical formatting describes document structure, such as paragraphs and headings, while physical formatting describes visual attributes like font sizes, colors, and spacings. With logical formatting, it would be easier to extract information than with the current physical formatting.

In general, the Web sites that are rich in information are built using "dynamic pages" that are generated on demand, in response to a user click or query. These pages are created by querying a local database, formatting the results as HTML, and then displaying such results to the user.

For example, let us consider a Web site about movies being shown in theaters. This Web site may present the movies on a tabular form with the titles, ratings, and show times, for instance. A Web search engine or other information provider interested on doing Web information extraction must read this table and reconstruct the original schema based on it. For example, it must find out that the first column contains the movie title, the second column the rating, and the third column the show times. This is easy for a human but it is hard to do it automatically. Most of the time, some information is lost, as depicted in Figure 23.5.

© 2012 Taylor & Francis Group, LLC

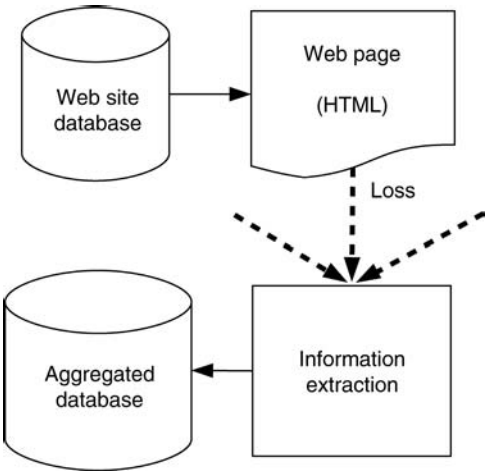

FIGURE 23.5 Information loss when extracting content from the Web.

Information extraction systems use clues from the page's formatting and structure, domain knowledge, and training examples, among other sources of information, to map HTML fragments to tuples in relations. They can also use methods for detecting the page template and isolating navigational areas that do not contribute content. The systems that do this task are informally known as "content scrapers" and they can be quite accurate, specially when restricted to particular domains. For a survey of information extraction methods, Kayed and Shaalan.[22]

Other aspects of content mining besides information extraction are content classification, sentiment analysis, and duplicated pages detection.

Content classification in general looks at statistics obtained from the Web pages to classify their contents. In many cases, this is done to find out what is the topic the contents are about. In other cases, content classification is used to extract document properties such as the genre of the document, or whether it expresses more opinions or more facts, or to evaluate how well-written a document is. In all cases, a statistical description of the document is created, and then a machine learning algorithm takes that description and a set of training labels to construct a model able to separate automatically the classes.[23,24]

Sentiment analysis, including "intention mining," is the task of finding what is the sentiment or intention of the author of a document. Specifically, it can be used to determine if a certain fragment is expressing a negative or positive opinion. This is very important given the large amount of product and service reviews available on Web pages, blogs, or forums. These reviews are typically very short, usually no more than a few paragraphs. The techniques of sentiment classification include the analysis of the frequency of certain terms,[25] with the aid of part-of-speech taggers or other natural language processing tools.

Finally, there is a significant amount of duplicate content on the Web. According to Broder, et al.,[26] roughly one-third of the pages on the Web are duplicates or near duplicates of another page, and recent studies have confirmed this trend. Finding near-duplicate content[27] is important for efficiency reasons, to avoid downloading and indexing many times the same pages. It is also important to filter out plagiarism, so that the original page gets ranked high, and not the copies.

Link Mining

The overall structure of the Web differs significantly from the one exhibited by random networks. The most salient difference is that, while on a random network most of the nodes have a degree

© 2012 Taylor & Francis Group, LLC

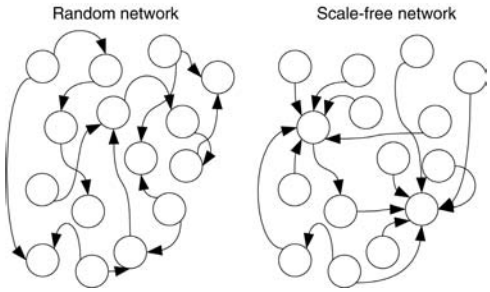

FIGURE 23.6 Difference between a random network and a scale-free network.

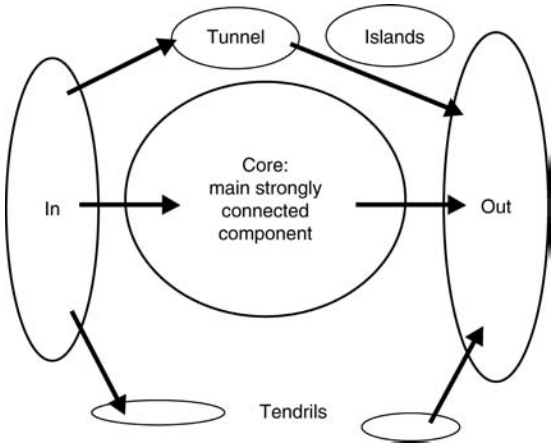

FIGURE 23.7 Bow-tie structure on the Web.

(number of connections) close to the average, in networks such as the Web, the distribution of the degree is very skewed. The networks that have this property are called scale-free networks.

Figure 23.6 depicts a random network and a scale-free network with the same number of nodes and edges. In a scale-free network, a few nodes attract more of the in-links. This can be explained by "rich-get-richer" processes[28] in which having many links gives a better chance of attracting new links, increasing the disparity in the number of connections over time.

At a macroscopic level, looking at the properties of the network as a whole, we can describe the Web in terms of the strongly connected components on it. A strongly connected component is a part of a graph in which all pairs of nodes can reach each other (in both directions) by following links. The Web exhibits a very large strongly connected component (CORE), other components that are reachable to/from it by directed links (IN and OUT, respectively), and pages that cannot be reached at all from the CORE, which are called ISLANDS. Minor components such as TENDRILS and TUNNEL can also be identified. This description is called the bow-tie structure of the Web[29] given its shape, depicted in Figure 23.7.

PageRank and HITS could be considered simple link mining techniques. More elaborated link analysis can be used for finding similar pages, communities, or detection of Web spam based in links.

Usage Mining

Usage data on the Web is abundant and valuable. Web site administrators can capture usage data by enabling logging on their Web servers, and they can enrich such data by instrumenting their internal

© 2012 Taylor & Francis Group, LLC

links. There are several free software packages available that can do sophisticated analysis of access logs and can discover, for instance, typical browsing paths. This is of particular importance for retailers and other e-commerce Web sites that can use this information to drive the design of their Web sites, improving the user experience and/or increasing their sales volume.

Search engines have access to the queries written by the users, and the pages they selected after seeing the list of results (and the pages they did not select). Data from user search sessions can be used to increase the relevance of the results.[30] Interesting relationships can be inferred by looking at users, queries, and pages. We can observe, for instance, that similar users tend to issue similar queries, that similar pages show up as results for related queries, and so on.

Usage data is increasingly valuable for search engines. Privacy issues arise in the confluence of the legal and technical aspects associated to this data collection, and both users and search engine have incentives for maintaining and enforcing the secrecy of this data.

CONCLUSIONS AND CURRENT TRENDS

As we have seen, Web retrieval methods differ from standard information retrieval methods, and can adapt to the large-scale, open, and distributed nature of the Web. For the future, two topics that are attracting a significant research effort are the mobile Web and the semantic Web.

The Mobile Web is the Web that is accessible and used through portable devices. Today, the capabilities of most mobile cell phones are well beyond just making phone calls. Many include Web-browsing software, and a growing fraction of the activity on the Web is carried through these devices, including browsing, searching, and even producing content (e.g., in the case of cell phones equipped with a camera). A challenge here is to provide users of portable devices with an experience that takes into account their geographical location and their current activity.

The Semantic Web[31] is a vision of the future of the Web, in which the Web contents can be read and understood by both humans and software agents. This will enable information integration and sharing without losing information. Several technologies enable the semantic Web, ranging from simple markup languages as the Extensible Markup Language (XML) to other languages that describe relationships among objects, classes, and properties. On top of these layers, applications will be able to analyze and, later, to reason about the contents and to extract knowledge from them.

REFERENCES

1. Arasu, A.; Cho, J.; Garcia-Molina, H.; Paepcke, A.; Raghavan, S. Searching the web. ACM Trans. Internet Technol. **2001**, *1* (1), 2–43.
2. Baeza-Yates, R.; Ribeiro-Neto, B. *Modern Information Retrieval*; Addison-Wesley: New York, 1999.
3. Chakrabarti, C. *Mining the Web: Analysis of Hypertext and Semi Structured Data*; Morgan Kaufmann: San Francisco, 2002.
4. Kobayashi, M.; Takeda, K. Information retrieval on the web. ACM Comput. Surv. **2000**, *32* (2), 144–173.
5. Koster, M. A standard for robot exclusion, http://www. robotstxt.org/wc/robots.html, 1996.
6. Ntoulas, A.; Cho, J.; Olston, C. What's new on the web?: The evolution of the web from a search engine perspective. In *Proceedings of the 13th conference on World Wide Web*; ACM Press: New York, 2004; 1–12.
7. Raghavan, S.; Garcia-Molina, H. Crawling the hidden web. In *Proceedings of the 27th International Conference on Very Large Data Bases*; Morgan Kaufmann: Rome, Italy, September 2001; 129–138.
8. Salton, G. *Introduction to Modern Information Retrieval (McGraw-Hill Computer Science Series)*; McGraw-Hill: New York, 1983.
9. Witten, I.H.; Moffat, A.; Bell, T.C. *Managing Gigabytes: Compressing and Indexing Documents and Images*; Morgan Kaufmann: San Francisco, 1999.
10. Tomasic, A.; Garcia-Molina, H. Performance of inverted indices in shared-nothing distributed text document information retrieval systems. In *PDIS '93: Proceedings of the Second International Conference on Parallel and Distributed Information Systems*, IEEE Computer Society Press: Los Alamitos, CA, 1993; 8–17.

© 2012 Taylor & Francis Group, LLC

11. Sahami, M.; Mittal, V.; Baluja, S.; Rowley, H. The happy searcher: Challenges in web information retrieval. In *8th Pacific Rim International Conference on Artificial Intelligence, volume 3157 of Lecture Notes in Computer Science*, Auckland, New Zealand, August 2004; Springer: Berlin/Heidelberg, 3–12.
12. Deerwester, S.; Dumais, S.T.; Furnas, G.W.; Landauer, T.K.; Harshman, R. Indexing by latent semantic analysis. J. Am. Soc. Inform. Sci. **1999**, *41* (6), 391–407.
13. Richardson, M.; Prakash, A.; Brill, E. Beyond pagerank: Machine learning for static ranking. In *Proceedings of the 15th international conference on World Wide Web*, ACM Press: Edinburgh, Scotland, May 2006; 707–715.
14. Davison, B.D. Topical locality in the web. In *Proceedings of the 23rd annual international ACM SIGIR Conference on Research and Development in Information Retrieval*, ACM: Athens, Greece, July 2000; 272–279.
15. Haas, S.W.; Grams, E.S. Page and link classifications: Connecting diverse resources. In *Proceedings of the third ACM conference on Digital libraries*, ACM Press: Pittsburgh, PA, June 1998; 99–107.
16. Borodin, A.; Roberts, G.O.; Rosenthal, J.S.; Tsaparas, P. Link analysis ranking: Algorithms, theory, and experiments. ACM Trans. Internet Technol. **2005**, *5* (1), 231–297.
17. Page, L.; Brin, S.; Motwani, R.; Winograd, T. The page-rank citation ranking: Bringing order to the Web, Technical report, Stanford Digital Library Technologies Project, 1998.
18. Kleinberg, J.M. Authoritative sources in a hyperlinked environment. J. ACM **1999**, *46* (5), 604–632.
19. Gori, M.; Witten, I. The bubble of web visibility. Commun. ACM **2005**, *48* (3), 115–117.
20. Gyöngyi, Z.; Garcia-Molina, H. Web spam taxonomy. In *First International Workshop on Adversarial Information Retrieval on the Web*, Chiba, Japan, May 2005; 39–47.
21. Henzinger, M.R.; Motwani, R.; Silverstein, C. Challenges in web search engines. SIGIR Forum **2002**, *36* (2), 11–22.
22. Kayed, M.; Shaalan, K.F. A survey of web information extraction systems. IEEE Trans. Know. Data Eng. **2006**, *18* (10), 1411–1428.
23. Dumais, S.; Chen, H. Hierarchical classification of web content. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, ACM Press: Athens, Greece, July 2000; 256–263.
24. Chakrabarti, S.; Dom, B.; Agrawal, R.; Raghavan, P. Scalable feature selection, classification and signature generation for organizing large text databases into hierarchical topic taxonomies. VLDB J. **1998**, *7* (3), 163–178.
25. Pang, B.; Lee, L.; Vaithyanathan, S. Thumbs up?: Sentiment classification using machine learning techniques. In *EMNLP '02*: Proceedings of the ACL-02 conference on Empirical methods in natural language processing, Association for Computational Linguistics: Philadelphia, PA, July 2002; 79–86.
26. Broder, A.Z.; Glassman, S.C.; Manasse, M.S.; Zweig, G. Syntactic clustering of the web. Comput. Netw. ISDN Syst. **1997**, *29* (813), 1157–1166.
27. Fetterly, D.; Manasse, M.; Najork, M. Detecting phrase-level duplication on the world wide web. In *SIGIR '05*: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval; ACM Press: New York, 2005; 170–177.
28. Barabási, A.L.; Albert, R. Emergence of scaling in random networks. Science **1999**, *286* (5439), 509–512.
29. Broder, A.; Kumar, R.; Maghoul, F.; Raghavan, P.; Rajagopalan, S.; Stata, R.; Tomkins, A.; Wiener, J. Graph structure in the web: Experiments and models. In *Proceedings of the Ninth Conference on World Wide Web*; ACM Press: Amsterdam, Netherlands, May 2000; 309–320.
30. Baeza-Yates, R. Applications of web query mining. In *Proceedings of the 27th European Conference on IR Research, ECIR 2005, volume 3408*; Springer: Santiago de Compostela, Spain, March 2005; 7–22.
31. Berners-Lee, T.; Hendler, J.; Lassila, O. The semantic web. Scientific American **2001**, *284* (5), 34–43.

BIBLIOGRAPHY

1. Information Retrieval methods, in general, and Web search, in particular, is discussed in the book by Baeza-Yates and Ribeiro-Neto [2, Chapter 13]. A textbook by Chakrabarti[3] deals with several topics related to Web Mining.

© 2012 Taylor & Francis Group, LLC

This page intentionally left blank

24 Semantic Web

Kieron O'Hara and Wendy Hall

CONTENTS

INTRODUCTION

The semantic web (SW) is an extension, in progress, to the World Wide Web (WWW), designed to allow software processes, in particular artificial agents, as well as human readers, to acquire, share, and reason about information. Whereas the WWW consists largely of documents, which are generally created for human consumption, the SW will be a web of data, making them more amenable for computers to process.[1] The data will be processed by computer via semantic theories for interpreting the symbols (hence: *semantic* web). In any particular application, the semantic theory will connect terms within a distributed document set logically, and thereby aid interoperability.

For instance, people use a lot of data in daily interactions, viewing bank statements, or digital photographs, or using diaries or calendars. But this does not constitute a web of data, because the data are neither exported from the applications in which they are stored or were created, nor linked to other relevant data. In a genuine web of data, such data could be used seamlessly in a number of applications. For example, one could view one's photographs (which will contain a time stamp) in

© 2012 Taylor & Francis Group, LLC

one's calendar, which would then act as a prompt to suggest what one was doing when they were taken. The data which one uses would be to some extent freed from the constraints of particular applications, and instead could be interlinked and reused creatively.

As another example, Web services can currently be accessed and executed via the Web, but because the Web does not provide much information-processing support, services must be specified using semiformal languages and as with information retrieval humans need to be kept in the loop. Web services described using SW techniques should provide support for autonomous agents and automatic systems.[2]

The world of linked information is a very unstructured, "scruffy" environment. The amounts of information that systems need to deal with are very large indeed. Furthermore, systems must pull together information from distributed sources, where representation schemes can be expected to be highly heterogeneous, information quality variable, and trust in information's provenance hard to establish. SW technology needs to be based on standards that can operate in this heterogenous information world.

The SW therefore requires two types of information standard to operate. First, it requires common formats for integrating information from these diverse sources. And second, it needs a language to express the mapping between the data and objects in the real world, in order to allow a seamless understanding of a distributed set of databases. Hence, for instance, we could signal that a database containing a column *zip code* and another database with a column labeled *ZC*, were actually both referring to the same concept with their different labels, and by creating such a semantic link, we could then start to reason over both databases in an integrated fashion. Such semantic links are often obvious to humans, but not to computers. A key formalism here is the *ontology*, which defines the concepts and relationships that we use in particular applications. Ontologies are central to the SW vision, as providing the chief means by which the terms used in data are understood in the wider context.[1,3]

THE AIM OF THE SEMANTIC WEB

The aim of the SW is to shift the emphasis of reasoning from documents to data, for three reasons. First, it will facilitate data reuse, often in new and unexpected contexts. Second, it will help reduce the amount of relatively expensive human information processing. Third, it will release the large quantity of information, not currently accessible, that is stored in relational databases (RDBs) by making it directly machine-processable.[4]

This implies that RDB objects must be exported to the Web as first-class objects, which in practice entails mapping them onto a consistent system of resource identifiers—called Universal Resource Identifiers (URIs—see below). The SW itself is a suite of languages and formalisms designed to enable the interrogation and manipulation of representations which make use of URIs.[1]

It is hoped that the SW will exhibit the same *network effects* that promoted the growth of the WWW. Network effects are positive feedback effects connected with *Metcalfe's law* that the value of a network is proportional to the square of the number of users/members. The more people share data that can be mapped onto URIs, the more valuable that data is. As value increases, more agents join the network to get the benefits, and include information that they own in the network which further increases its value. This, like the WWW model, is radically different from other models of the value of information, wherein value is dictated by *scarcity* (copyright, intellectual property restrictions, etc). In decentralized networks like the Web the value of information is dictated by *abundance*, so it can be placed in new contexts, and reused in unanticipated ways.

This is the dynamic that enabled the WWW to spread, when the value of Web documents was seen to be greater in information-rich contexts. One initiative to support the development of the SW is the creation of a discipline of *web science*, which is intended to exploit study of both technical and social issues to predict such matters with more accuracy.[5,6]

If the SW is to grow in an analogous way, more data has to be exposed to the Web that can be mapped onto URIs. In practice, this means that the data must be exposed in the resource description

© 2012 Taylor & Francis Group, LLC

framework (RDF), an agreed international standard whose role in the SW is described below[7]; in particular, it can be used not only to assert a link between two resources, but also to name (and therefore make explicit) the relationship that links them. RDF is the language of choice for reuse, because it is a relatively inexpressive language compared to other formalisms used in the SW (see Figure 24.1 for a pictorial representation of the layers of formalisms required for the SW vision—expressivity increases as we ascend the diagram). The importance of RDF in this model is dictated by the so-called principle of least power, which states that the less expressive the representation language, the more reusable the data.[8]

The importance of growth is such that a stage can be reached when reuse of data—one's own or that of other people—is facilitated. There would ideally be so much information exposed in RDF that the contexts into which one's own data can be placed would be rich enough and numerous enough to increase its value significantly. RDF (as described below) represents information as a subject–predicate–object triple each of whose component parts is a URI. If the objects, resources, or representations referred to by the URIs are defined in ontologies, then this enables the interoperability at which the SW aims.

Hence another vital component in the SW is the development and maintenance of ontologies. These must be endorsed by the communities that use them, whether they are large-scale, expensive ontologies developed as a result of a major research effort, or relatively ad hoc creations intended to support small-scale collaboration.

Ontologies can also play an important role in bringing (representatives of) two or more communities together for a common purpose, by expressing a common vocabulary for their collaboration, onto which the terms of each discipline can be mapped. Such collaborative efforts are extremely important for reuse of content.[3]

This is not to say that search and retrieval on the current Web is not of high quality; the methods pioneered by Google and others work very well. Nevertheless, keyword-based search techniques are vulnerable to a number of well-known flaws. Individual words can be ambiguous. A document can refer to a topic of interest without using the keyword. Keywords are language-dependent. Information distributed across several documents cannot be amalgamated by keyword search. And even though PageRank and related algorithms for search produce impressive results, the user still

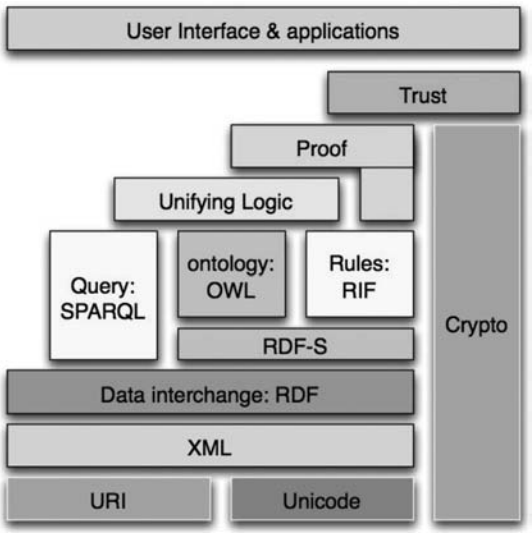

FIGURE 24.1 The layered view of the semantic web. (From A framework for Web Science, by T. Berners-Lee, W. Hall, J.A. Hendler, K. O'Hara, N. Shadbolt, D.J. Weitzner, Found. Trends Web Sci. 2006, *1* (1), 1–134.)

© 2012 Taylor & Francis Group, LLC

needs to read manually through the ordered list of retrieved pages, and inspect their content to determine relevance to his/her inquiry. This involvement of the user is a hindrance to scalability.

The SW should make more accurate querying possible, using ontologies to help with problems of ambiguity and unused keywords, and data linking to query across distributed datasets. Furthermore, it should be able to go beyond current search with respect to the three issues of reuse, automation, and exploitation of RDBs. And as well as search and retrieval, the addition of information processing support to the Web will help promote other functions such as Web services and knowledge management.

COMPONENTS OF THE SEMANTIC WEB

At one level, the SW is a complex of formalisms and languages each doing a different job in the representation of information, as shown in Figure 24.1. Each formalism is an internationally agreed standard (see below), and the composition of the functions these formalisms serve supports semantically enabled reasoning on data.

At the bottom of this diagram stands the URIs which identify the resources about which the SW provides reasoning capabilities.[9] The universality of URIs is extremely important—i.e., it is vital that whatever naming convention is used for URIs is adopted globally, so as to create the network effects that allow the SW to add value. Interpretation of URIs must also be consistent across contexts. In other words, when we *dereference* URIs (i.e., when we locate the resource to which the URI refers), we should always get the same object. If these conditions about URI naming schemes are met, then making an association between a URI and a resource means that different people can refer or link to it consistently in their conversations. The other basic formalism, Unicode, is an industry standard that allows computers to represent text in different writing systems.

The next layer up, eXtensible Markup Language (XML), is a language to mark up documents, and a uniform data exchange format between applications.[10] It allows the insertion of user-defined tags into documents that provide information about the role that the content plays. So, for instance, XML allows one to write a document describing a book, and also to *annotate* the document with machine-readable *metadata* to indicate e.g., who the authors of the book are.

RDF[7] is a very minimal knowledge representation framework for the Web, which uses a basic subject–predicate–object structure, with the twist that it assigns specific URIs to its individual fields—including in the predicate position, thereby identifying a relationship between the entities identified by the connected nodes. This use of URIs allows us to reason not only about objects but also about the relationships between them. XML is a metalanguage that provides a uniform framework for markup, but it does not provide any way of getting at the *semantics* of data; RDF is the first step toward semantics.

The resource description framework schema (RDFS, sometimes known as RDF(S)[11]) gives greater scope for sharing information about individual domains; whereas RDF is a data interchange language that lets users describe resources using their own vocabularies, and makes no assumptions about the domains in question, RDFS provides a basic set of tools for producing structured vocabularies that allow different users to agree on particular uses of terms. An extension of RDF, it adds a few modeling primitives with a fixed meaning (such as class, subclass and property relations, and domain and range restriction).

A key component for SW applications is the *ontology*. Ontologies[3] are shared conceptualizations of a domain which are intended to facilitate knowledge and information sharing by coordinating vocabulary and allowing basic inference of inheritance and attributes of objects. Several initiatives are developing ontologies, particularly in a number of sciences, which means that the scientists are likely to be among the important early adopters of SW technology (see below). RDFS is an important step toward the SW vision, as the addition of modeling primitives makes it a basic ontology representation language.

© 2012 Taylor & Francis Group, LLC

However, greater expressivity is likely to be required in the development of more complex ontologies, and the World Wide Web Consortium (W3C) has issued a Web Ontology Language (OWL[12]) in multiple versions that allows ontologies to be not only represented but also checked for logical properties such as consistency. The three species of OWL are: 1) OWL Full, containing all the OWL primitives, allowing arbitrary combination of those primitives with RDF and RDFS (allowing changes in meaning even of predefined OWL or RDF primitives), but also providing so much expressive power as to make the language undecidable (i.e., it cannot be guaranteed that a computation using the full expressive power of OWL Full will be completed in a finite time); 2) OWL DL, which restricts application of OWL's constructors to each other, and corresponds to a decidable *description logic*, but which is not fully compatible with RDF; and 3) OWL Lite, which sacrifices even more expressive power to facilitate implementation and reasoning.[12] This set of relations affects the downward compatibility of the SW layer diagram—the only version of OWL that is downward compatible with RDF and RDFS (i.e., so that any processor for that version of OWL will also provide correct interpretations of RDFS) is OWL Full, which is undecidable (pp. 113–115).[13,14]

All varieties of OWL use RDF for their syntax, and use the linking capabilities of RDF to allow ontologies to be distributed—ontologies can refer to terms in other ontologies. Such distributivity is a key property for an ontology language designed for the SW.[15]

OWL supports some kinds of inference, such as subsumption and classification, but a greater variety of rules and inference is needed. Hence, work is currently ongoing on the Rule Interchange Format (RIF), which is intended to allow a variety of rule-based formalisms, including Horn-clause logics, higher order logics, and production systems, to be used.[16] Various insights from Artificial Intelligence (AI) have also been adapted for use for the SW, including temporal (time-based) logic, causal logic, and probabilistic logics.[1]

Having represented data using RDF and ontologies, and provided for inference, it is also important to provide reliable, standardized access to data held in RDF. To that end, a special query language SPARQL (pronounced "sparkle"), which became a W3C recommendation in January 2008, has been designed.[17] Logic and proof systems are envisaged to sit on top of these formalisms, to manipulate the information in deployed systems.[1]

A very important layer is that of *trust*.[18] If information is being gathered from heterogeneous sources and inferred over, then it is important that users are able to trust such sources. The extent of trust will of course depend on the criticality of the inferences—trust entails risk, and a risk-averse user will naturally trust fewer sources.[19,20] Measuring trust, however, is a complex issue.[21] A key parameter is that of provenance, a statement of: 1) the conditions under which; 2) the methods with which; and 3) the organization by which, data were produced. Methods are appearing to enable provenance to be established, but relatively little is known about how information spreads across the Web.[22]

Related issues include respect for intellectual property, and the privacy of data subjects. In each case the reasoning abilities of the SW can be of value, and initiatives are currently under way to try to exploit them.[23] Creative commons[24] is a way of representing copyright policies and preferences based on RDF to promote reuse where possible (current standard copyright assumptions are more restrictive with respect to reuse). And research into the policy aware web is attempting to develop protocols to allow users to express their own privacy policies, and to enable those who wish to use information to reason about those policies.[25] Cryptography protocols to protect information will also play an important role, as shown in Figure 24.1.

ADDITIONAL FACTORS IN SEMANTIC WEB DEVELOPMENT

Infrastructure

Another important part of SW development is the infrastructure that supports it. In particular, if data is to be routinely published to the Web in RDF format, there must be information repositories that can store RDF and RDFS. These *triple stores* (so-called because they store the RDF

© 2012 Taylor & Francis Group, LLC

triples) must provide reasoning capabilities as well as retrieval mechanisms, but importantly must be *scalable*. Examples of triple stores include JENA,[26] 3store,[27,28] and Oracle 11*g*.[29] OWLIM is a repository which works as a storage and inference layer for the Sesame RDF database, providing reasoning support for some of the more expressive languages of the SW, RDFS, and a limited version of OWL Lite.[30,31]

Reasoners

As representation in the SW is more complex than in previous technologies, so is reasoning. The area of SW reasoning has been the focus of much research, in order to infer the consequences of a set of assertions interpreted via an ontology. In such a context, inference rules need clear semantics, and need to be able to cope with the diverse and distributed nature of the SW.

There are a number of important issues of relevance in this area: 1) Under what conditions is negation monotonic (i.e., the addition of new facts does not change the derivation of not-p), or nonmonotonic (including negation as failure, deriving not-p from the failure to prove p)?; 2) How should we handle conflicts when merging rule-sets?; 3) "Truth" on the Web is often dependent on context—how should a reasoner represent that dependence?; 4) How should scalability be balanced against expressivity?; 5) Logic often assumes a static world of given "facts," but how should it be adapted to the SW, a much more dynamic space where propositions are asserted and withdrawn all the time?; and 6) The heterogeneous nature of the SW means that data in the SW is of varying trustworthiness; how should a reasoner deal with variable reliability? None of these questions has a "correct" answer, but any SW reasoning system needs to address them.

There has been a lot of research on SW reasoning, but an important desideratum is that a reasoner should support the W3C recommended formalisms, in particular supporting OWL entailment at as high a level as possible, and SPARQL querying. Examples include: Jena, an open source SW framework for Java, with a rule-based inference engine[32]; Pellet, a sound and complete OWL-DL reasoner[33]; and KAON2, an infrastructure for managing ontologies written in OWL-DL and other SW rule languages.[34] For a short review of the problems and prospects for SW reasoning, see Fensel.[35]

Bootstrapping

Bootstrapping content for the SW is one more important issue. Sufficient content is required for the hoped-for network effects to appear. There are initiatives to generate data in RDF and to expose it on the Web as a vital first step. The DBpedia[36] is based on the Web 2.0 community-created encyclopedia Wikipedia, and is intended to extract structured information from Wikipedia allowing much more sophisticated querying. Sample queries given on the DPpedia Web site include a list of people influenced by Friedrich Nietzsche, and the set of images of American guitarists. DBpedia uses RDF, and is also interlinked with other data sources on the Web. When accessed in late 2007, the DBpedia dataset consisted of 103 million RDF triples. Other examples of linked data applications include the DBLP bibliography of scientific papers,[37] and the GeoNames database which gives descriptions of millions of geographical features in RDF.[38]

Even if RDF began to be published routinely, there is still a great deal of legacy content on the Web, and to make this accessible to SW technology some automation of the translation process is required. Gleaning Resource Descriptions from Dialects of Languages (GRDDL) allows the extraction of RDF from XML documents using transformations expressed in Extensible Stylesheet Language Transformations (XSLT) an extensible stylesheet language based on XML. It is hoped that such extraction could allow bootstrapping of some of the hoped-for SW network effects.[39]

Annotating documents and data with metadata about content, provenance, and other useful dimensions (even including relevant emotional reactions to content[40]) is also important for the effort to bring more content into the range of SW technologies.[41] Multimedia documents, such as

© 2012 Taylor & Francis Group, LLC

images, particularly benefit from such annotation.[42] Again, given the quantities of both legacy data, and new data being created, methods of automating annotation have been investigated by a number of research teams in order to increase the quantity of annotated data available without excessive expenditure of resources.[41,43,44]

The Social Context: Web Science

The SW vision has been delineated with some care by the W3C, and as has been seen involves an intricate set of connections between a number of formalisms, each of which is designed to do a certain job. As we will describe in the next section, that vision has altered and gained complexity over time.

In general, there are severe complications in the mapping between the microlevel engineering of Web protocols, and the macrolevel social effects that result from large-scale use of the Web. The combination of scales, effects, and phenomena involved is too large to be easily covered by a single discipline, even computer science. The social interactions enabled by the Web place demands on the Web applications underlying them, which in turn put requirements on the Web's infrastructure. However, these multiple requirements are not currently well-understood.[45] Social studies tend to regard the Web as a given, whereas the Web is rather a world changeable by alterations to the protocols underlying it. Furthermore, the Web changes at a rate that is at least equal and may be faster than our ability to observe and analyze it.

The SW is a development bringing the Web vision to a new level of abstraction, yet the current state of our knowledge of the Web and its relation to off-line society leaves a number of questions unanswered about how it will impact at a large scale. In particular, it is unknown what social consequences there might be of the greater public exposure and sharing of information that is currently locked in databases. Understanding these consequences is important partly because the developers of the SW want to build a technology that is not harmful to society thanks to emergent social effects, and partly because it is important that the SW goes with the grain of society, in order that it be effective in real-world situations.[5]

To this end, in 2006 the Web Science Research Initiative (WSRI) was set up as a joint venture by the Massachusetts Institute of Technology and the University of Southampton to foster the interdisciplinary study of the Web in its social and technical context. WSRI's role includes crafting a curriculum for study across the various relevant disciplines; Berners-Lee.[6] is a detailed review of the wide range of scientific and social-scientific research that is likely to be relevant, including graph and network theory, computer science, economics, complexity theory, psychology, law, etc.

HISTORY AND INTELLECTUAL BACKGROUND

The vision of a web of data was always implicit in the ideas underlying the development of the WWW, and was articulated by Sir Tim Berners-Lee at the first WWW conference in 1994. Berners-Lee is well known as the inventor of the WWW in 1989–1991, and has been a leading figure in the development of the SW. As well as holding chairs at the Massachusetts Institute of Technology, United States, and the University of Southampton, United Kingdom, Berners-Lee is the director of the W3C, which he founded in 1994.

A key moment in the development, and public perception, of the SW was an entry written for *Scientific American* by Berners-Lee, James A. Hendler, and Ora Lassila in 2001.[46] This entry postulated the next stage of the WWW explicitly as one where data and information, as well as documents, are processed automatically, and envisaged a world where intelligent agents were able to access information (e.g., from calendars, gazetteers, and business organizations) in order to undertake tasks and planning for their owners.

This vision of automation of a series of routine information processing tasks has not emerged at the time of writing (2008). The article's agent-oriented vision distracted attention from the main

© 2012 Taylor & Francis Group, LLC

point of the SW, the potential of a web of linked *data* (as opposed to documents) with shared semantics. Hence, in 2006, Berners-Lee, together with Nigel Shadbolt and Wendy Hall, published another article in the IEEE journal *Intelligent Systems*, which made that point explicitly, and argued that the agent-based vision would only flourish with well-established data standards.[1]

The *Scientific American* article painted a very enticing picture, but its key message was less to do with the agents and more to do with the semantic information infrastructure that Berners-Lee et al. were advocating. Indeed, the infrastructure will be used for many knowledge management purposes, not only in allowing agents to communicate. The agent-focused rhetoric of the article has prompted some to argue that the SW is a restatement of the program of AI in the 1960s and 1970s, and will share its perceived failures. We address this question below, in the section entitled "Controversies."

In 2001 (and before), the conceptualization of the various formal layers of the SW was as shown in Figure 24.2, with a fairly straightforward cascade up from URIs to XML and namespaces, to RDF and RDFS, through ontologies to rules, logic, proof and trust (the diagram has been widely distributed, but see e.g., Berners-Lee).[47] Comparison with Figure 24.1 shows how the details of the SW layers have had to be amended over time as implementation has continued. The requirements for expression of ontology-related information has led to an extra complexity from that envisaged in 2001, while the criticism of the SW vision based on the *Scientific American* article has led to a realization that not only the expressive formalisms need to be in place, but also tools and methods need to be created to allow use of SW technologies to integrate smoothly into organizations' standard information workflows (e.g., Shadbolt, Vargas-Vera, Golbeck and Alani[1,44,48,49]). This led to a top layer, User Interface, being added to the Figure 24.2 structure at a later date.

Where intelligent agency has appeared—and there are currently several applications, including shopbots and auction bots—it has tended to be handcrafted and unable to interact with heterogeneous information types. This is largely because of a lack of well-established scalable standards for information sharing; however, progress is being made toward that goal, especially via the painstaking committee-based standards development processes instituted by the W3C. These standards are crucial for the SW to "take off," and for the hoped-for network effects of a large number of users to emerge.[1]

The SW vision has been implemented by standard bodies, such as the Internet Engineering Task Force (IETF) as well as the W3C (the W3C is responsible for standards specific to the WWW), which have orchestrated efforts together with the user community to develop the languages at various levels to share meaning. Once standards are set by the W3C, they are called *recommendations*, acknowledging the reality that with the decentralization of the Web, and a lack of a central authority, standards cannot be enforced. The first RDF standard was specified in 1997 and became a W3C recommendation in 1999, thereby providing a minimal knowledge representation language for the Web with the clear backing of the nascent SW community.

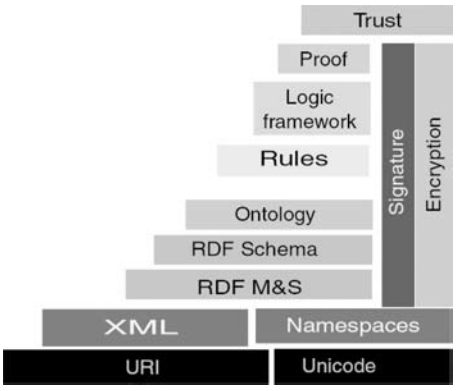

FIGURE 24.2 The early layered view of the Semantic Web.

© 2012 Taylor & Francis Group, LLC

Fixed standards for expressing ontologies appeared later in the process, with RDFS and OWL becoming recommendations in 2004. OWL evolved from other ontology language efforts, including Ontology Inference Layer (OIL)[50] and DARPA Agent Markup Language (DAML)[51] whose merged product, DAML+OIL, was the most important predecessor to OWL.[52] In January 2008, the query language SPARQL became a W3C recommendation, while the RIF was under development in mid-2008.

Figure 24.3, created in 2003, illustrates Berners-Lee's vision of the pattern of SW development using the visual metaphor of a tide flowing onto a beach (this diagram is widely available, but see Connolly).[53] From top to bottom in the diagram are the various layers of the SW diagram, from trust and proof down to data exchange and markup. From left to right come the various stages in a rough lifecycle from research to deployment: the first stage is a blue-sky research project; the second is the production of a stable system or formalism that is not a standard; the best aspects of these systems are then used as the bases for W3C standards, and the final stage is one of wide deployment. Hence, for instance, early ontology efforts like Cyc and description logics led to efforts such as DAML and OIL, which in turn helped create OWL. Wide deployment of OWL then results in a so-called web of meaning.

The "sea" of research and deployment approaches from the bottom left of Figure 24.3 to the top right, as the "tide" comes in. Hence in 1998, various formalisms were in place for all the various levels of representation of the SW, but only XML was a Web standard and beginning to be used widely. By 2003, OWL and RDFS were close to their final forms, and RDF was beginning to be used widely for cross-application interoperability. At the time of writing, the "tide" has advanced further to the right, so work is ongoing on rule language RIF, and query language SPARQL became an official W3C recommendation in 2008. Meanwhile OWL is being used more frequently by ontology builders.

The SW's history to date is largely one of standard-setting. However, it has also been argued that, analogous to other systems which have spread quickly and grown exponentially, what is needed is a "killer app" (i.e., an application that will meet a felt need and create a perception of the technology

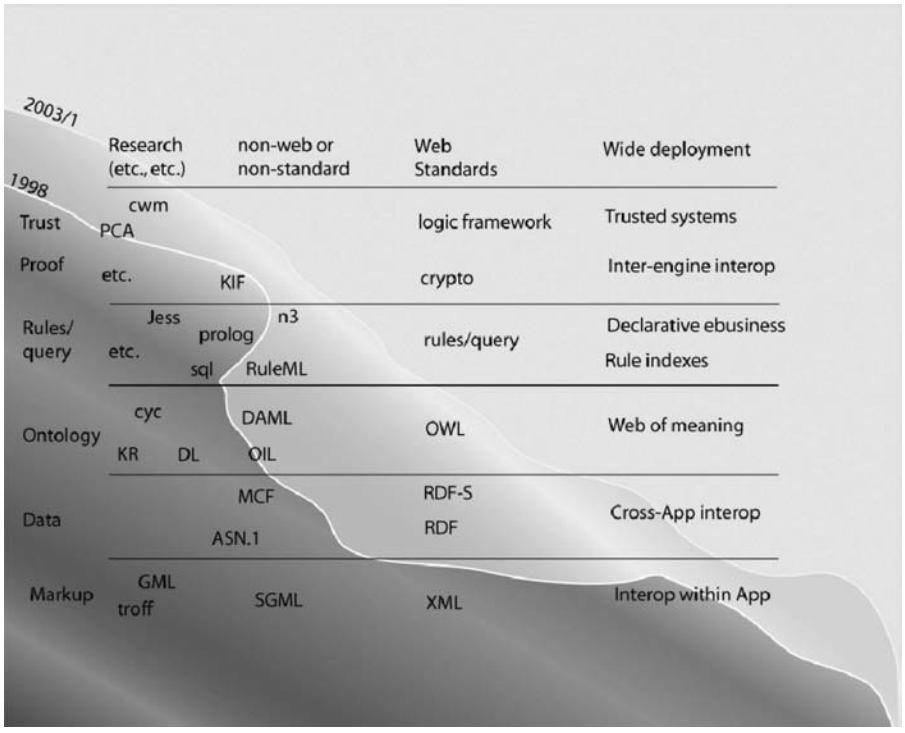

FIGURE 24.3 A representation of the progress of semantic web development.

© 2012 Taylor & Francis Group, LLC

as "essential"). Less ambitiously, the SW's spread depends not only on having an impressive set of formalisms, but also software tools to use information represented in those formalisms.[49] The SW is clearly not, at the time of writing, an information resource in routine use. Nevertheless, there are some applications where SW technologies are serving valuable purposes, and we review some of these in the next section.

APPLICATIONS AND SYSTEMS

Properties of Systems

In general, SW projects tend to exhibit a few constant features. They generate new ontologies for the application domain (for example, art, or computer science), and use them to interrogate large stores of data, which could be legacy data or freshly harvested. Hence a body of evidence is building up that ontologies have an important role in mediating the integration of data from heterogeneous sources.

Furthermore, the results of SW projects are generally presented using custom-built interfaces. This hints at a very important research area, which is the development of scalable visualizers capable of navigating the graph of connected information expressed in RDF. As can be seen, the importance of applications and user interfaces was made clear in the latest version of the layered SW diagram (Figure 24.1).

In this section we will look at active SW successes, focusing on application areas and types, then commercial/real-world systems, before finally looking at some of the more successful academic efforts as judged by the SW development community itself.

Application Areas

There are areas where the SW is already an important tool, often in small focused communities with pressing information-processing requirements and various more or less common goals. Such communities can function as early adopters of the technology, exactly as the high energy physics discipline played a vital role in the development of the WWW. A series of case studies and use cases is maintained in w3.org's Web site.[54]

The most important application for SW technology is *e-science*, the data-driven, computationally intensive pursuit of science in highly distributed computational environments.[55] Very large quantities of data are created by analyses and experiments in disciplines such as particle physics, meteorology, and the life sciences. Furthermore, in many contexts, different communities of scientists will be working in an interdisciplinary manner, which means that data from various fields (e.g., genomics, clinical drug trials, and epidemiology) need to be integrated. Many accounts of distinct and complex systems (e.g., the human body, the environment) consist of data brought from disciplines varying not only in vocabulary, but also in the scale of description; understanding such systems, and the way in which events at the microscale affect the macroscale and *vice versa*, is clearly an important imperative. Many scientific disciplines have devoted resources to the creation of large-scale and robust ontologies for this and other purposes. The most well-known of these is the *gene ontology*, a controlled vocabulary to describe gene and gene product attributes in organisms, and related vocabularies developed by open biomedical ontologies.[56] Others include the protein ontology, the cell cycle ontology, Medical Subject Headings (MeSH, used to index life science publications), systematized nomenclature of medicine (SNOMED), and AGROVOC (agriculture, forestry, fisheries, and food).

E-government is another potentially important application area, where information is deployed widely, and yet is highly heterogeneous. Government information varies in provenance, confidentiality, and "shelf life" (some information will be good for decades or even centuries, while other information can be out-of-date within hours), while it can also have been created by various levels

© 2012 Taylor & Francis Group, LLC

of government (national/federal, regional, state, city, and parish). Integrating that information in a timely way is clearly an important challenge (see for instance a pilot study for the United Kingdom's Office of Public Sector Information, exploring the use of SW technologies for disseminating, sharing, and reusing data held in the public sector.[57])

Commercial Activity

There is an increasing number of applications that allow a deeper querying of linked data. We have already discussed DBpedia,[36] DBLP,[37] and GeoNames.[38] Commercial applications are also beginning to appear. Garlik[58] is a company seeking to exploit SW-style technologies to provide individual consumers with more power over their digital data. It reviews what is held about people, harvesting data from the open Web, and represents this in a people-centric structure. Natural Language Processing is used to find occurrences of people's names, sensitive information, and relations to other individuals and organizations. Declaration of interest: Wendy Hall is chair of the Garlik Advisory Board. Twine[59] is intended to enable people to share knowledge and information, and to organize that information using various SW technologies (also, like Garlik, using Natural Language Processing). Twine's developer Nova Spivack has coined the term "knowledge networking" to describe the process, analogous to the Web 2.0 idea of "social networking."

The increasing maturity of SW technology is being shown by the growing number of successful vendors of SW technology. We have already seen OWLIM,[31] which was developed by Ontotext, a semantic technology lab focused on technologies to support the SW and SW services based in Sofia, Bulgaria, and Montreal, Canada; Ontotext has been and is a partner in a number of major SW research projects.[60] Ontoprise, based in Karlsruhe, Germany, is a software vendor for implementing SW infrastructure in large, distributed enterprises; its products include OntoBroker, which provides ontology support using the W3C recommended languages OWL, RDFS and SPARQL, and Semantic MediaWiki+, a collaborative knowledge management tool.[61] Asemantics, with offices in Italy, Holland, and the United Kingdom, uses a combination of Web 2.0 paradigms with SW technologies such as XML and RDF. The SW technologies are powerful representational tools but are often perceived as hard to use and search, so Asemantics attempts to exploit the perceived usability of Web 2.0 to present data in more widely accepted formats.[62]

Academic Work: The Semantic Web Challenge

Much of the major work in the SW has been carried out in the academic sphere, and in funded research projects between academic and commercial partners, and is reported in journals and conferences (see end of entry for a list of the more importance conferences). Any review of academic work in this field will inevitably be selective; for the purposes of this entry we will focus on a particular effort to nurture applications, the *Semantic Web Challenge*.

The SW Challenge was created in 2003, and associated with the International Semantic Web Conference (ISWC) of that year. Since then it has become an annual competition to create an application that shows SW technology in its best aspects, and which can act as a "benchmark" application. Hence the SW Challenge gives us a series of illustrative applications thought by researchers' peers to constitute best SW practice.[63]

To meet the criteria for the Challenge, a tool or system needs to meet a number of requirements,[64] which provide a useful characterization of the expectations governing an SW system, and are suggestive of the expected properties of SW applications. For instance, it should use information from sources that are distributed and heterogeneous, of real-world complexity and with diverse ownership. It should assume an open world, and that the information is never complete, and it should use some formal description of the meaning of the data. Optional criteria include a use of data in some way other than the creators intended, use of multimedia, and use of devices other than a PC. Applications need not be restricted to information retrieval, and ideally the system would be

© 2012 Taylor & Francis Group, LLC

scalable in terms of the amount of data used and the number of distributed components cooperating. All these criteria indicate areas where SW systems would be expected to have an advantage.

The winners of the SW Challenge to date are as follows:

- 2003: CS AKTive Space (University of Southampton), an integrated application which provides a way to explore the U.K. Computer Science Research domain across multiple dimensions for multiple stakeholders, from funding agencies to individual researchers, using information harvested from the Web, and mediated through an ontology.[65]
- 2004: Flink (Vrije Universiteit Amsterdam), a "Who's Who" of the SW which allows the interrogation of information gathered automatically from Web-accessible resources about researchers who have participated in ISWC conferences.[66]
- 2005: CONFOTO (appmosphere Web applications, Germany), a browsing and annotation service for conference photographs.[67]
- 2006: MultimediaN E-Culture Demonstrator (Vrije Universiteit Amsterdam, Centre for Mathematics and Computer Science, Universiteit van Amsterdam, Digital Heritage Netherlands and Technical University of Eindhoven), an application to search, navigate, and annotate annotated media collections interactively, using collections from several museums and art repositories.[68]
- 2007: Revyu.com (Open University), a reviewing and rating site specifically designed for the SW, allowing reviews to be integrated and interlinked with data from other sources (in particular, other reviews).[69]

CONTROVERSIES

The SW vision has always generated controversy, with a number of commentators being highly skeptical of its prospects. Let us briefly review some of the disputed issues.

The Semantic Web as "Good Old-Fashioned Artificial Intelligence"

One view holds that the SW is basically a throwback to the project to program machine intelligence which was jokingly christened by John Haugeland "GOFAI" (good old-fashioned AI). This proved impossible: so much of human intelligence is implicit and situated that it was too hard a problem to write down everything a computer needed to know to produce an output that exhibited human-like intelligence. For instance, if a human is told about a room, further explanations that a room generally has a floor, at least three walls, usually four, and a ceiling, and some method of ingress that is generally but not always a door, are not required. But a computer needs to be told these mundane facts explicitly—and similarly every time it is introduced to a new concept.[70]

One attempt to work around this problem is the Cyc project, set up in 1984, which aims to produce a gigantic ontology that will encode all commonsense knowledge of the type about the room given above, in order to support human-like reasoning by machines.[71] The project has always aroused controversy, but it is fair to say that over two decades later, GOFAI is no nearer. The implicit nature of commonsense knowledge arguably makes it impossible to write it all down.

Many commentators have argued that the SW is basically a re-creation of the (misconceived) GOFAI idea, that the aim is to create machine intelligence over the Web, to allow machines to reason about Web content in such a way as to exhibit intelligence.[72,73] This, however, is a misconception, possibly abetted by the strong focus in the 2001 *Scientific American* article on an agent-based vision of the SW.[46] Like many GOFAI projects, the scenarios in that article have prominent planning components. There is also continuity between the AI tradition of work on formal knowledge representation and the SW project of developing ontologies (see below).

The SW has less to do with GOFAI as with context-based machine reasoning over content (and the provision of machine-readable data on the Web). The aim is not to bring a single ontology, such

© 2012 Taylor & Francis Group, LLC

as Cyc, to bear on all problems (and therefore implicitly to define or anticipate all problems and points of view in the ontology definition), but rather to allow data to be interrogated in ways that were not anticipated by their creators. Different ontologies will be appropriate for different purposes; composite ontologies can be assembled from distributed parts (thanks to the design of OWL); and it is frequently very basic ontologies (defining simple terms such as "customer," "account number," or "account balance") that deliver large amounts of content. It is, after all, a matter of fact that people from different communities and disciplines can and do interact without making any kind of common *global* ontological commitment.[1,6,74]

Indeed, we can perhaps learn from the experience of hype and reaction that accompanied the development of AI. There has been a great deal of criticism of AI, but much has been learned from AI research and some AI methods and systems are now routinely exploited in a number of applications. The same may be expected of the SW. We should not expect to wake up one morning with the SW implemented and ready for use. Rather, a likelier model is that SW technologies will be incorporated into more systems "behind the scenes" wherever methods are needed to deal with signature SW problems (large quantities of distributed heterogeneous data).

Arguments for and Against Ontologies

The importance of ontologies for the SW has been another point of friction with those who believe the program unrealistic. Ontologies are seen as expensive to develop and hard to maintain. Classification of objects is usually done relative to some task, and as the nature of the task changes, ontologies can become outdated. Classifications are also made relative to some background assumptions, and impose those assumptions onto the resulting ontology. To that extent, the expensive development of ontologies reflects the world view of the ontology builders, not necessarily the users. They are top-down and authoritarian, and therefore opposed to the Web ethos of decentralization and open conversation. They are fixed in advance, and so they don't work very well to represent knowledge in dynamic, situated contexts.[75–77]

Furthermore, say the critics, the whole point of the Web as a decentralized, linked information structure is that it reflects the needs of its large, heterogeneous user base which includes very many people who are naïve in their interactions. The infrastructure has to be usable by such people, which argues for simplicity. The rich linking structure of the current Web, combined with statistically based search engines such as Google, is much more responsive to the needs of unsophisticated users. The SW, in contrast, demands new information markup practices, and corporations and information owners need to invest in new technologies. Not only that, but current statistical methods will scale up as the number of users and interactions grows, whereas logic-based methods such as those advocated by the SW, on the other hand, scale less well (cf., e.g., Zambonini).[78]

Folksonomies

One development as part of the so-called Web 2.0 paradigm (of systems, communities, and services which facilitate collaboration and information-sharing among users) that has drawn attention in this context is that of the "folksonomy." Folksonomies have arisen out of the recent move to allow users to "tag" content on Web 2.0 sites such as the image-sharing site Flickr, and the video-sharing site YouTube. Having seen content, users are allowed to tag it with key words, which, when the number of users has become large enough, results in a structure of connections and classifications emerging without central control. Their promoters argue that folksonomies "really" express the needs of their users (since all the structure has arisen out of their user-based classifications), whereas ontologies "really" express the needs of authorities who can "impose" their views from the top-down.[76]

However, folksonomies are much less expressive than ontologies; they are basically variants on keyword searches. A tag "SF" may refer to a piece of science fiction, or to San Francisco, or something else from the user's private idiolect. Indeed, that ambiguity arises even if we make the

© 2012 Taylor & Francis Group, LLC

unrealistic assumption of a monoglot English user community. Once we realize speakers of other languages will use a system, then there are further possible ambiguities—for instance, in German "SF" might refer to the Swiss television station Schweizer Fernsehen.

Resolving This Controversy

When a community is large enough and the benefits clear, then a large-scale ontology building and maintenance program is justified. In a recent note, Berners-Lee argues that such conditions will be perhaps more frequently encountered than skeptics believe. On the very broad assumptions that the size of an ontology-building team increases as the order of the log of the size of the ontology's user community, and that the resources needed to build an ontology increase as the order of the square of community size, the cost per individual of ontology building will diminish rapidly as user community size increases. Of course these assumptions are not intended to be deeply realistic, so much as indicative of how the resource implications diminish as the community increases in size. Berners-Lee's moral: "Do your bit. Others will do theirs."[74]

Even so, not all ontologies need to be of great size and expressive depth. Certainly the claim that has been made that the SW requires a single ontology of all discourse on the model of Cyc, but this is not backed up by the SW community. Such an ontology, even if possible, would not scale, and in a decentralized structure like the Web its use could not be enforced. We should rather expect a lot of use of small-scale, *shallow* ontologies defining just a few terms that nevertheless are widely applicable.[74] Experience in building real-world SW systems often shows that expectations about the cost and complexity of the ontologies required are overblown, and the ontology-building process can be relatively straightforward and cheap.[79]

For example, the machine-readable friend-of-a-friend (FOAF) ontology is intended to describe people, their activities, and their relations to other people. It is not massively complex, and indeed publishing a FOAF account of oneself is a fairly simple matter of form-filling (using the FOAF-a-matic tool).[80] But the resulting network of people (showing their connections to other people) has become very large indeed. A survey performed in 2004 discovered over 1.5 million documents using the FOAF ontology.[81]

With respect to Folksonomies, it is important to note that ontologies and folksonomies serve different purposes. Folksonomies are based on word tags, whereas the basis for ontology reference is via a URI. One of the main aims of ontology definition is to *remove* ambiguity—not globally, for this may well be impossible, but rather within the particular context envisaged by the developer (see the section on "Symbol Grounding" below). Folksonomies will necessarily inherit the ambiguity of the natural language upon which they are based. And while folksonomies emerge from data-sharing practices, it is not necessarily the case the ontologies are authoritarian; rather, the latter should ideally be *rationalizations* of current sharing practice. This does entail departure from current practice, but not necessarily of great magnitude. Indeed, a strong possibility is to use cheaply gathered folksonomies as starting points for ontology development, gradually morphing the Web 2.0 structures into something with greater precision and less ambiguity.[82]

Symbol Grounding

An important aspect of the SW is that URIs must be interpreted consistently. However, terms and symbols are highly variable in their definitions and use through time and space. The SW project ideally needs processes whereby URIs are given to objects, such that the management of these processes is by communities and individuals, endorsed by the user community, who ensure consistency. This URI "ownership" is critical to the smooth functioning of the SW.[1]

But the process of *symbol grounding* (i.e., ensuring a fixed and known link between a symbol and its referent) is at best hard, and at worst (as argued by Wittgenstein, for instance) impossible.[83,84] Meanings do not stay fixed, but alter, often imperceptibly. They are delineated not only by traditional

© 2012 Taylor & Francis Group, LLC

methods such as the provision of necessary and sufficient conditions, but also by procedures, technologies and instrumentation, and alter subtly as practice alters.

Any attempt to fix the reference of URIs is a special case of symbol grounding, and is consequently hard to do globally. It is certainly the case that attempting to resist the alteration in community practices and norms, and reformulation of meanings of terms, would be doomed.

Yorick Wilks has argued that since much knowledge is held in unstructured form, in plain text, automatic Natural Language Processing techniques, statistically based, can be used to "ground" meanings of terms for the SW.[73] Berners-Lee, on the other hand, maintains that the SW is necessarily based on logic and firm definitions (even if those definitions were imperfect, or highly situated and task-relative), not words, use patterns and statistics. Wilks' point is that the aim of defining terms in logic is too idealistic, and anyway depends on false assumptions about ordinary word meaning. Berners-Lee's counterargument is, in effect, that though meanings are not stable, they can be stable *enough* relative to individual applications and in particular contexts to allow the SW approach to work.

CONCLUSION

The SW has been somewhat misunderstood in some commentaries. Its aim is not to force users to accept large ontologies remote from data-sharing practice imposed by shadowy authorities. Neither is it intended to produce a theory of all discourse, or to reproduce GOFAI. Rather, it is intended to shift the emphasis of the Web from being a web of documents to a web of linked *data*. It is the development of formalisms and technologies facilitating the creation, sharing, and querying of linked data using sharable ontologies to establish common interpretations. For this reason, an alternative name for the SW is the *web of linked data*.

The SW is a work in progress. As it stands, the "buy in" to the SW has not yet produced the desirable network effects, although several disciplines are enthusiastic early adopters of the technology (e.g., the e-science community). And there are still several important research issues outstanding. It is not yet known how best to: 1) query large numbers of heterogeneous information stores at many different scales; 2) translate between, merge, prune, or evaluate ontologies; 3) visualize the SW; and 4) establish trust and provenance of the content.

As complex technologies and information infrastructures are developed, there is a dynamic feedback between requirements, analysis, engineering solutions, and hard-to-predict global behavior of human, machine, and hybrid systems. Understanding how basic engineering protocols governing how computers talk to each other can result in social movements at a very different level of abstraction is very hard, yet essential to realizing the SW vision. Indeed, such understanding, the defining purpose of the discipline of *Web Science*, is essential to ensuring that *any* Web-based information structure is beneficial.[5]

ACKNOWLEDGMENTS

The authors would like to thank Tim Berners-Lee, Nigel Shadbolt, James A. Hendler, Daniel J. Weitzner, Harith Alani, Marcia J. Bates, and an anonymous referee for helpful comments and discussions.

REFERENCES

1. Shadbolt, N.; Hall, W.; Berners-Lee, T. The Semantic Web revisited. IEEE Intell. Syst. **2006**, *21* (3), 96–101.
2. Fensel, D.; Bussler, C.; Ding, Y.; Kartseva, V.; Klein, M.; Korotkiy, M.; Omelayenko, B.; Siebes, R. Semantic Web application areas. In *7th International Workshop on Applications of Natural Language to Information Systems (NLDB 2002)*, Stockholm, Sweden, June 27–28, 2002, http://www. cs.vu.nl/~ronny/work/NLDB02.pdf, 2002 (accessed July 2008).

© 2012 Taylor & Francis Group, LLC

3. Fensel, D. *Ontologies: A Silver Bullet for Knowledge Management and Electronic Commerce*, 2nd Ed.; Springer: Berlin, 2004.
4. Berners-Lee, T. Relational databases on the Semantic Web, http://www.w3.org/DesignIssues/RDB-RDF.html, 1998 (accessed December 2007).
5. Berners-Lee, T.; Hall, W.; Hendler, J.; Shadbolt, N.; Weitzner, D. Creating a science of the Web. Science **2006**, *313* (5788), 769–771.
6. Berners-Lee, T.; Hall, W.; Hendler, J.A.; O'Hara, K.; Shadbolt, N.; Weitzner, D.J. A framework for Web Science. Found. Trends Web Sci **2006**, *1* (1), 1–134.
7. Klyne, G.; Carroll, J.J.; McBride, B. Resource Description Framework (RDF): Concepts and abstract syntax, 2004 http://www.w3.org/TR/rdf-concepts/ (accessed December 2007).
8. Berners-Lee, T. Principles of design, 1998, http://www.w3.org/DesignIssues/Principles.html (accessed December 2007).
9. Berners-Lee, T.; Fielding, R.; Masinter, L. Uniform Resource Identifier (URI): Generic syntax, 2005, http://gbiv.com/proto-cols/uri/rfc/rfc3986.html (accessed December 2007).
10. Bray, T.; Paoli, J.; Sperberg-McQueen, C.M.; Maler, E.; Yergeau, F. *Extensible Markup Language (XML) 1.0*, 4th Ed.; 2006, http://www.w3.org/TR/xml/ (accessed December 2007).
11. Brickley, D.; Guha, R.V.; McBride, B. RDF vocabulary description language 1.0: RDF Schema, 2004, http://www.w3.org/TR/rdf-schema/ (accessed December 2007).
12. McGuinness, D.L.; van Harmelen, F. OWL Web Ontology Language overview, 2004, http://www.w3.org/TR/owl-features/ (accessed December 2007).
13. Antoniou, G.; van Harmelen, F. *A Semantic Web Primer*; MIT Press: Cambridge MA, 2004.
14. Dean, M.; Schreiber, G.; Bechhofer, S.; van Harmelen, F.; Hendler, J.; Horrocks, I.; McGuinness, D.L.; Patel-Schneider, P.F.; Stein, L.A. OWL Web Ontology Language Reference, 2004, http://www.w3.org/TR/owl-ref/ (accessed December 2007).
15. Smith, M.K.; Welty, C.; McGuiness, D.L. OWL Web Ontology Language guide, 2004, http://www.w3.org/TR/owl-guide/ (accessed December 2007).
16. Boley, H.; Kifer, M. RIF basic logic dialect, 2007, http:// www.w3.org/TR/rif-bld/ (accessed December 2007).
17. Prud'hommeaux, E.; Seaborne, A. SPARQL query language for RDF, 2007, http://www.w3.org/TR/rdf-sparql-query/ (accessed December 2007).
18. Golbeck, J. Trust on the World Wide Web: A survey. Found. Trends Web Sci. **2006**, *1* (2), 1–72.
19. Bonatti, P.A.; Duma, C.; Fuchs, N.; Nejdl, W.; Olmedilla, D.; Peer, J.; Shahmehri, N. Semantic Web policies—a discussion of requirements and research issues. In *The Semantic Web: Research and Applications*, 3rd European Semantic Web Conference 2006 (ESWC-06), Budva, Montenegro, 2006; Sure, Y., Domingue, J., Eds.; Springer: Berlin, 2006.
20. O'Hara, K.; Alani, H.; Kalfoglou, Y.; Shadbolt, N. Trust strategies for the Semantic Web. In *Workshop on Trust, Security and Reputation on the Semantic Web*, 3rd International Semantic Web Conference (ISWC 04), Hiroshima, Japan, 2004, http://eprints.ecs.soton.ac.uk/10029/ (accessed December 2007).
21. Golbeck, J.; Hendler, J. Accuracy of metrics for inferring trust and reputation in Semantic Web-based social networks. In *Engineering Knowledge in the Age of the Semantic Web*, Proceedings of 14th International Conference, EKAW 2004, Whittlebury Hall, U.K. 2004; Motta, E.; Shadbolt, N.; Stutt, A.; Gibbins, N.; Eds.; Springer: Berlin, 2004; 116–131.
22. Groth, P.; Jiang, S.; Miles, S.; Munroe, S.; Tan, V.; Tsasakou, S.; Moreau, L. An architecture for provenance systems, http://eprints.ecs.soton.ac.uk/13216/1/ provenanceArchitecture10.pdf, 2006 (accessed December 2007).
23. O'Hara, K.; Shadbolt, N. *The Spy in the Coffee Machine: The End of Privacy As We Know It*; Oneworld: Oxford, 2008.
24. http://creativecommons.org/about/ (accessed December 2007).
25. Weitzner, D.J.; Hendler, J.; Berners-Lee, T.; Connolly, D. Creating a policy-aware Web: Discretionary, rule-based access for the World Wide Web. In *Web and Information Security*, Ferrari, E.; Thuraisingham, B.; Eds.; Idea Group Inc: Hershey, PA, 2005.
26. http://jena.sourceforge.net/ (accessed December 2007).
27. http://sourceforge.net/projects/threestore (accessed December 2007).
28. Harris, S.; Gibbins, N. 3store: Efficient bulk RDF storage. In *Proceedings of the 1st International Workshop on Practical and Scalable Systems*, Sanibel Island, FL, 2003, http://km.aifb.uni-karlsruhe.de/ws/psss03/proceedings/harris-et-al.pdf (accessed December 2007).
29. http://www.oracle.com/technology/tech/semantic_technolo gies/index.html (accessed December 2007).
30. http://www.ontotext.com/owlim/ (accessed July 2008).

© 2012 Taylor & Francis Group, LLC

31. Kiryakov, A.; Ognyanov, D.; Manov, D. OWLIM: A pragmatic semantic repository for OWL. In *Web Information and Systems Engineering–WISE 2005 Workshops*, Proceedings of the Workshop on Scalable Semantic Web Knowledge Base Systems at WISE 2005, New York, November 2005; Dean, M.; Guo, Y.; Jun, W.; Kaschek, R.; Krishnaswamy, S.; Pan, Z.; Sheng, Q.Z.; Eds.; Springer: Berlin, 2005; 182–192, http://www.ontotext.com/publica tions/ssws_owlim.pdf (accessed July 2008).
32. McBride, B. Jena: Implementing the RDF model and syntax specification. In *Proceedings of the 2nd International Workshop on the Semantic Web: SemWeb 2001*, World Wide Web Conference 2001, Hong Kong, May 2001; Decker, S.; Fensel, D.; Sheth, A.; Staab, S.; Eds.; CEUR-WS, Vol. 40, 2001, http:// sunsite.informatik.rwth-aachen. de/Publications/CEUR-WS/Vol-40/mcbride.pdf (accessed July 2008).
33. Sirin, E.; Parsia, B.; Cuenca Grau, B.; Kalyanpur, A.; Katz, Y. Pellet: A practical OWL-DL reasoner. J. Web Semant. **2007**, *5* (2), 51–53.
34. http://kaon2.semanticweb.org/ (accessed July 2008).
35. Fensel, D.; Van Harmelen, F. Unifying reasoning and search to Web scale. IEEE Internet Comput. **2007**, *11* (2), 96, 94–95 (sic).
36. Auer, S.; Bizer, C.; Kobilarov, G.; Lehmann, J.; Cyganiak, R.; Ives, Z. DBpedia: A nucleus for a Web of open data. In *Proceedings of the 6th International Semantic Web Conference 2007*, Busan, South Korea, 2007, http://iswc2007. semanticweb.org/papers/715.pdf (accessed December 2007).
37. http://www4.wiwiss.fu-berlin.de/dblp/ (accessed December 2007).
38. http://www.geonames.org/ (accessed December 2007).
39. Connolly, D., Ed. Gleaning Resource Descriptions from Dialects of Languages (GRDDL), 2007, http:// www.w3. org/TR/grddl/ (accessed December 2007).
40. Schröder, M.; Zovato, E.; Pirker, H.; Peter, C.; Burkhardt, F. W3C emotion incubator group report, 2007, http://www.w3.org/2005/Incubator/emotion/XGR-emotion/ (accessed December 2007).
41. Handschuh, S.; Staab, S.; Eds. *Annotation for the Semantic Web*; IOS Press: Amsterdam, 2003.
42. Troncy, R.; van Ossenbruggen, J.; Pan, J.Z.; Stamou, G.; Halaschek-Wiener, C.; Simou, N.; Tsouvaras, V. Image annotation on the Semantic Web, 2007, http://www.w3.org/ 2005/Incubator/mmsem/XGR-image-annotation/ (accessed December 2007).
43. Handschuh, S.; Staab, S.; Ciravegna, F. S-CREAM—Semi-automatic Creation of Metadata. In *Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web*, Proceedings of 13th International Conference, EKAW 2002, Siguënza, Spain, 2002; Gómez-Pérez, A.; Benjamins, V.R.; Eds.; Springer: Berlin, 2002; 358–372.
44. Vargas-Vera, M.; Motta, E.; Domingue, J.; Lanzoni, M.; Stutt, A.; Ciravegna, F. MnM: Ontology-driven semiautomatic and automatic support for semantic markup. In *Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web*, Proceedings of 13th International Conference, EKAW 2002, Siguënza, Spain, 2002; Gómez-Pérez, A.; Benjamins, V.R.; Eds.; Springer: Berlin, 2002; 379–391.
45. Hendler, J.; Shadbolt, N.; Hall, W.; Berners-Lee, T.; Weitzner, D. Web Science: An interdisciplinary approach to understanding the World Wide Web. Commun. ACM **2008**, *51* (7), 60–69.
46. Berners-Lee, T.; Hendler, J.; Lassila, O. The Semantic Web. Sci. Am. May **2001**, http://www.sciam.com/ article. cfm?articleID=00048144-10D2-1C70-84A9809EC588EF21 (accessed December 2007).
47. Berners-Lee, T. Foreword. In *Spinning the Semantic Web: Bringing the World Wide Web to its Full Potential*; Fensel, D.; Hendler, J.; Lieberman, H.; Wahlster, W.; Eds.; MIT Press: Cambridge, MA, 2003; xi–xxiii.
48. Golbeck, J.; Grove, M.; Parsia, B.; Kalyanpur, A.; Hendler, J. New tools for the Semantic Web. In *Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web*, Proceedings of 13th International Conference, EKAW 2002, Siguënza, Spain, 2002; Gómez-Pérez, A.; Benjamins, V.R.; Eds.; Springer: Berlin, 2002; 392–400.
49. Alani, H.; Kalfoglou, Y.; O'Hara, K.; Shadbolt, N. Towards a killer app for the Semantic Web. In *The Semantic Web*, Proceedings of the International Semantic Web Conference 2005, Hiroshima, Japan, 2005; Gil, Y.; Motta, E.; Benjamins, V.R.; Musen, M.A.; Eds.; Springer: Berlin, 2005; 829–843.
50. Fensel, D.; Horrocks, I.; van Harmelen, F.; Decker, S.; Erdmann, M.; Klein, M. OIL in a nutshell. In *Knowledge Engineering and Knowledge Management: Methods, Models and Tools*, Proceedings of 12th European Knowledge Acquisition Workshop (EKAW 2000), Juan-les-Pins, France, October 2000; Dieng, R.; Corby, O.; Eds.; Springer: Berlin, 2000; 1–16, http://www.cs.vu.nl/~onto-know/oil/downl/ oilnutshell.pdf (accessed July 2008).
51. http://www.daml.org/about.html (accessed July 2008).
52. Patel-Schneider, P.; Horrocks, I.; van Harmelen, F. Reviewing the design of DAML + OIL: An ontology language for the Semantic Web. In *Proceedings of the 18th National Conference on Artificial Intelligence*

© 2012 Taylor & Francis Group, LLC

(AAAI02), Edmonton, Canada, 2002, http://www.cs.vu.nl/~frankh/postscript/AAAI02.pdf (accessed December 2007).
53. Connolly, D. Semantic Web update: OWL and beyond, 2003, http://www.w3.org/2003/Talks/1017-swup/all.htm (accessed December 2007).
54. http://www.w3.org/2001/sw/sweo/public/UseCases/ (accessed December 2007).
55. Hendler, J.; de Roure, D. E-science: The grid and the Semantic Web. IEEE Intell. Syst. **2004**, *19* (1), 65–71.
56. http://www.geneontology.org/ (accessed July 2008).
57. Alani, H.; Dupplaw, D.; Sheridan, J.; O'Hara, K.; Darlington, J.; Shadbolt, N.; Tullo, C. Unlocking the potential of public sector information with Semantic Web technology. In *Proceedings of the 6th International Semantic Web Conference 2007*, Busan, South Korea, 2007, http://iswc2007.semanticweb .org/papers/701.pdf (accessed December 2007).
58. https://www.garlik.com/index.php (accessed December 2007).
59. http://www.twine.com/ (accessed December 2007).
60. http://www.ontotext.com/index.html (accessed July 2008).
61. http://www.ontoprise.de/index.php?id=134 (accessed July 2008).
62. http://www.asemantics.com/index.html (accessed July 2008).
63. http://www.informatik.uni-bremen.de/agki/www/swc/index. html (accessed December 2007).
64. http://challenge.semanticweb.org/ (accessed December 2007).
65. Schraefel, M.M.C; Shadbolt, N.R.; Gibbins, N.; Glaser, H.; Harris, S. CS AKTive Space: Representing computer science on the Semantic Web. In *Proceedings of WWW 2004*; New York, 2004, http://eprints .ecs.soton.ac.uk/9084/ (accessed December 2007).
66. Mika, P. Flink: Semantic Web technology for the extraction and analysis of social networks. J. Web Semant. **2005**, *3* (2), http://www.websemanticsjournal.org/papers/20050719/ document7.pdf (accessed December 2007).
67. Nowack, B. CONFOTO: A semantic browsing and annotation service for conference photos. In *The Semantic Web*, Proceedings of the International Semantic Web Conference 2005, Hiroshima, Japan, 2005; Gil, Y.; Motta, E.; Benjamins, V.R.; Musen, M.A.; Eds.; Springer: Berlin, 2005; 1067–1070.
68. Schreiber, G.; Amin, A.; van Assem, M.; de Boer, V.; Hardman, L.; Hildebrand, M.; Hollink, L.; Huang, Z.; van Kersen, J.; de Niet, M.; Omelayenko, B.; van Ossenbruggen, J.; Siebes, R.; Taekema, J.; Wielemaker, J.; Wielinga, B. MultimediaN e-culture demonstrator, 2006, http://www.cs.vu.nl/~guus/papers/Schreiber06a.pdf (accessed December 2007).
69. Heath, T.; Motta, E. Revyu.com: A reviewing and rating site for the Web of data. In *Proceedings of the 6th International Semantic Web Conference 2007*, Busan, South Korea, 2007, http://iswc2007.semantic-web.org/papers/889. pdf (accessed December 2007).
70. Haugeland, J. Understanding natural language. J. Philos. **1979**, *76*, 619–632.
71. Lenat, D.B. Cyc: A large-scale investment in knowledge infrastrcture. Commun. ACM **1995**, *38* (11), 32–38.
72. Jones, K.S. What's new about the Semantic Web? Some questions. SIGIR Forum **2004**, *38* (2), http://www.sigir. org/forum/2004D/sparck jones_sigirforum_2004d.pdf (accessed December 2007).
73. Wilks, Y. The Semantic Web: Apotheosis of annotation, but what are its semantics? IEEE Intell. Syst. **2008**, *23* (3), 41–49.
74. Berners-Lee, T. The fractal nature of the Web, 2007, http://www.w3.org/DesignIssues/Fractal.html (accessed December 2007).
75. Pike, W.; Gahegan, M. Beyond ontologies: Toward situated representations of scientific knowledge. Intl. J. Hum. Comput. Stud. **2007**, *65* (7), 674–688.
76. Shirky, C. Ontology is overrated: categories, links and tags, 2005, http://www.shirky.com/writings/ontology_o-verrated.html (accessed December 2007).
77. Stevens, R.; Egaña Aranguren, M.; Wolstencroft, K.; Sattler, U.; Drummond, N.; Horridge, M.; Rector, A. Using OWL to model biological knowledge. Intl. J. Hum. Comput. Stud. **2007**, *65* (7), 583–594.
78. Zambonini, D. The 7 (f)laws of the Semantic Web, 2006, http://www.oreillynet.com/xml/blog/2006/06/the_7_flaws_of_the_semantic_we.html (accessed December 2007).
79. Alani, H.; Chandler, P.; Hall, W.; O'Hara, K.; Shadbolt, N.; Szomsor, M. Building a pragmatic Semantic Web. IEEE Intell. Syst. **2008**, *23* (3), 61–68.
80. http://www.ldodds.com/foaf/foaf-a-matic (accessed December 2007).
81. Ding, L.; Zhou, L.; Finin, T.; Joshi, A. How the Semantic Web is being used: An analysis of FOAF documents. In *Proceedings of the 38th International Conference on System Sciences, 2005*, http://ebiquity .umbc.edu/_file_ directory_/papers/120.pdf (accessed December 2007).

© 2012 Taylor & Francis Group, LLC

82. Mika, P. Ontologies are us: A unified model of social networks and semantics. J. Web Semant. **2007**, *5* (1), 5–15.
83. Harnad, S. The symbol grounding problem. Physica D **1990**, *42*, 335–346. http://users.ecs.soton.ac.uk/harnad/Papers/ Harnad/harnad90.sgproblem.html (accessed December 2007).
84. Wittgenstein, L. *Philosophical Investigations*; Basil Blackwell: Oxford, 1953.

BIBLIOGRAPHY

1. Antoniou, G.; van Harmelen, F. *A Semantic Web Primer*; MIT Press: Cambridge MA, 2004.
2. Berners-Lee, T. *Weaving the Web: The Past, Present and Future of the World Wide Web by Its Inventor*; Texere Publishing: London, 1999.
3. Berners-Lee, T.; Hall, W.; Hendler, J.A.; O'Hara, K.; Shadbolt, N.; Weitzner, D.J. A framework for web science. Found. Trends Web Sci. **2006**, *1* (1), 1–134.
4. Berners-Lee, T.; Hall, W.; Hendler, J.; Shadbolt, N.; Weitzner, D. Creating a science of the Web. Science **2006**, *313* (5788), 769–771.
5. Berners-Lee, T.; Hendler, J.; Lassila, O. The Semantic Web. Sci. Am. May **2001**. http://www.sciam.com/article. cfm?articleID=00048144-10D2-1C70-84A9809EC588EF21 (accessed December 2007).
6. Fensel, D. *Ontologies: A Silver Bullet for Knowledge Management and Electronic Commerce*, 2nd Ed.; Springer: Berlin, 2004.
7. Fensel, D.; Hendler, J.; Lieberman, H.; Wahlster, W. *Spinning the Semantic Web: Bringing the World Wide Web to its Full Potential*; MIT Press: Cambridge, MA, 2003.
8. Shadbolt, N.; Hall, W.; Berners-Lee, T. The Semantic Web revisited. IEEE Intell. Syst. **2006**, *21* (3), 96–101.
9. There are several important annual conferences for the SW community, including: the World Wide Web Conference (WWW); the International Semantic Web Conference (ISWC—pronounced Iss-wick); the European Semantic Web Conference. These conferences preserve their proceedings online.
10. The World Wide Web Consortium's Semantic Web activity page is at http://www.w3.org/2001/sw/, and contains references to interviews, manifestos and statements by key SW developers. It also maintains a useful site of case studies and use cases at http://www.w3.org/2001/sw/sweo/ public/UseCases/. For Web Science, see http://webscience.org/.

© 2012 Taylor & Francis Group, LLC

This page intentionally left blank

25 XML Information Retrieval

Mounia Lalmas

CONTENTS

INTRODUCTION

Documents can be structured or unstructured. Unstructured documents have no (or very little) fixed predefined format, whereas structured documents are usually organized according to a fixed predefined structure. An example of a structured document is a book organized into chapters, each with sections made of paragraphs and so on. Nowadays, the most common way to format structured content is with the W3C standard (http://www.w3.org/XML/) for information repositories and exchanges, the eXtensible Mark-up Language (XML).

Much of the content available on the Web is formatted in HTML. Although HTML imposes some structure on a Web content, this structure is mainly for presentation purposes and carries little meaning. In contrast, XML is used to provide meaning about the stored content. More precisely, in the context of text documents, with which this entry is concerned, XML is used to specify the logical, or tree, structure of documents, in which separate document parts (e.g., chapter, section, abstract) and their logical structure (e.g., a chapter made of sections, a section and its title, an article and its abstract) are explicitly marked-up. As an increasing number of documents are being made available in XML format, effective means to access them are needed. As for standard (unstructured) documents, this requires appropriate query languages, representation methods, and ranking algorithms.

© 2012 Taylor & Francis Group, LLC

Approaches for accessing logically structured documents were first proposed in the 1990s.[1–4] In the late 1990s, as XML was adopted as the standard document format, approaches for what became known as XML information retrieval were being developed.[5–7] Research in XML information retrieval was then further boosted with the set-up in 2002 of the Initiative for the Evaluation of XML Retrieval (INEX),[8] a yearly evaluation campaign that provides a forum for the evaluation of approaches specifically developed for XML information retrieval. INEX provides test collections and evaluation measures, which make it possible for organizations worldwide to evaluate and compare their XML information retrieval approaches.

By exploiting the logical structure of XML documents, the goal of an XML information retrieval system is to implement so-called focused retrieval strategies, which aim at returning document components, i.e., XML elements, instead of whole documents in response to a user query. These focused retrieval strategies aim to break away from the traditional retrieval unit of a document as a single large (text) block. This is believed to be of particular benefit for information repositories containing long documents, or documents covering a wide variety of topics (e.g., books, user manuals, legal documents), where the users effort to locate relevant content within a document can be reduced by directing them to the most useful parts, i.e., the most useful XML elements, in the document.

To identify the most useful XML elements to return as answers to given queries, XML information retrieval systems require:

- Query languages that allow users to specify the nature of relevant components, in particular with respect to their structure
- Representation strategies providing a description not only of the content of XML documents, but also their structure
- Ranking strategies that determine the most relevant elements and rank these appropriately for a given query

In this entry, we provide an overview of "Query Languages," "Representation Strategies," and "Ranking Strategies" developed for XML information retrieval. The representation and ranking strategies presented in this entry were evaluated within the INEX evaluation campaigns.[9–14] The entry finishes with some conclusions on XML information retrieval research, and some references to early work related to XML information retrieval.

QUERY LANGUAGES

XML documents are organized into a logical structure, as provided by the XML mark-up. For example, a scientific article, such as those forming the IEEE test collection used in INEX (see Figure 25.1), consists of a front matter (`<fm>`), a body (`<bdy>`), and a back matter (`<bm>`). The front matter contains the article's metadata, such as title, author, publication information, and abstract. Following it is the article's body, which contains the actual content of the articles, and is structured into sections (`<sec>`), subsections (`<ss1>`), and sub-sub-sections (`<ss2>`). These logical units start with a title, followed by a number of paragraphs. The back matter contains a bibliography and further information about the article's authors.

Users may want to specify conditions to limit the search to specific XML elements. For example, a user may want sections discussing "XML retrieval evaluation," whereas another user may look for paragraphs about "effectiveness measures" contained in sections about "XML retrieval evaluation." Here we have a combination of content constraints, "XML retrieval evaluation" and "effectiveness measures," typical to information retrieval, and structural constraints, "section," "paragraph," and "paragraph within section." XML query languages have been developed with the aim to express various levels of content and structural constraints. They can be classified as content-only or content-and-structure query languages.

© 2012 Taylor & Francis Group, LLC

```
<article>
  <fm>
  ...
   <ti>IEEE Transactions on...</ti>
     <atl>Construction of...</atl>
     <au>
        <fnm>John</fnm>
        <snm>Smith</snm>
        <aff>University of...</aff>
     </au>
     <au>...</au>
  ...
   </fm>
   <bdy>
     <sec>
        <st>Introduction</st>
        <p>...</p>
     ...
   </sec>
   <sec>
     <st>...</st>
     ...
     <ss1>...</ss1>
     <ss1>...</ss1>
   ...
   </sec>
   ...
  </bdy>
  <bm>
   <bib>
     ...
   </bib>
</bm>
</article>
```

FIGURE 25.1 Sketch of the structure of a typical article in the INEX test collection.

Content-only queries make use of content constraints only, i.e., they are made of words, which is the standard form of input in information retrieval. They are suitable for XML retrieval scenarios where users do not know, or are not concerned, with the document logical structure when expressing their information needs. Although only content conditions are being specified, XML information retrieval systems must still determine what are the best fragments, i.e., the XML elements at the most appropriate level of granularity, to return to satisfy these conditions. For example, the best answer for a query "XML retrieval evaluation" may be a subsection and not a section, as the section, although relevant, may be less specific to the query than the subsection. An XML information retrieval system task is to determine this appropriate level of granularity for any given query.

Content-and-structure query languages provide a means for users to specify content and structural information needs. It is toward the development of this type of queries that most research on XML query languages lies. We can distinguish between three main categories of content-and-structure XML query languages, namely in sections "Tag-Based Languages," "Path-Based Languages," and "Clause-Based Languages." For the latter two types, we provide a brief description, mainly through examples, of current languages, namely XPath and Narrowed Extended XPath I (NEXI), and XQuery and XQuery Full-Text, respectively.

Tag-Based Queries

With tag-based queries, words in the query are annotated with a single tag name, which specifies the type of desired result elements, e.g., a section, an abstract. For example, the information

© 2012 Taylor & Francis Group, LLC

need "retrieve sections about XML retrieval evaluation" would be expressed as `section:XML retrieval evaluation`. An example of a tag-based query language is XSEarch.[15]

Tag-based queries are intuitive, and have been used in domains outside XML information retrieval (e.g., faceted search, Web search). However they only express simple, although important and likely common, structural constraints. They cannot express, for instance, relationship (structural) constraints, e.g., "a paragraph contained in a section," which may be needed for complex retrieval scenarios.

Path-Based Queries

Path-based queries are based upon the syntax of XPath (XML Path language, http://www.w3.org/TR/xpath), which has been defined by the W3C to navigate to components of an XML document. The most important concept in XPath is the location path, which consists of a series of navigation steps characterizing movements within an XML document.

For example, `chapter/section` is a location path, where `chapter` and `section` are steps that navigate to elements of types "chapter" and "section," respectively. The fact that the steps are separated by "/" means that the location path selects section elements directly below chapter elements. Section elements are referred to as children of chapter elements. The navigation steps can be separated by "//". For example, `chapter//section` navigates to all section elements that are directly or indirectly below a chapter element. Section elements are referred to as descendants of chapter elements. Special steps include the self step denoted "." and parent step "..". For example, `.// section` returns all section elements contained directly or indirectly in the currently navigated element.

At each step, predicates can be specified between "[" and "]", which must be satisfied for elements to be navigated into. For example, the following XPath query `//article[@year=2002]/title` selects the "titles" of "articles" published in 2002, and only those.

An important function in XPath for the purpose of XML information retrieval is the function `contains()`. For example, the query `//section [fn : contains (. /title, "XML retrieval")]` will return all section elements with a title containing the string "XML retrieval". The result of this XPath query is a set of section elements, and not a ranked list of section elements. Thus XPath is not an XML query language that can be directly used in XML information retrieval. Nonetheless, it is used by, or has inspired, other path-based query languages, some of which allowing the ranking of results, e.g., XXL,[16] XIRQL,[17] and NEXI.[18] We discuss the last one, NEXI.

The NEXI query language was developed by INEX, as a simple query language for XML information retrieval evaluation. NEXI consists of a small but enhanced subset of XPath. The enhancement comes from the introduction of a new function, named `about()`, which requires an element to be about some specified content criteria. It replaces the XPath `contains()` function, to reflect that an element can be relevant to a given query without actually containing any of the words used in the query.

A small subset of XPath was chosen because NEXI was not developed to test the expressiveness of a query language for XML information retrieval, but to evaluate XML information retrieval effectiveness. For instance, the parent/child navigation step "/" was considered particularly problematic as it was open to misinterpretation by assessors, and hence was dropped. We recall that in information retrieval evaluation, assessors are used in the process of building a test collection. Their task is to judge the relevance of the information returned to them as answers to given queries. All result elements must have at least one `about()` function. This is because for the purpose of evaluating retrieval effectiveness, what matters is that the relevant elements are actually returned. For instance, the following query `//section [about(., XML retrieval)]//title`, which requests titles of sections about "XML retrieval evaluation," is not allowed in NEXI; it is indeed a mechanical process to return the title of a section deemed relevant to "XML retrieval."

© 2012 Taylor & Francis Group, LLC

We finish with an example of a NEXI query:

```
//article [about(. //bdy, XML retrieval)]//
section [about(., evaluation)]
```

This query is asking for section elements about "evaluation" contained in articles that have a body that discusses "XML retrieval."

NEXI was developed by INEX for the purpose of evaluating XML information retrieval effectiveness. It remains the task of the XML information retrieval system to interpret a NEXI query, where the interpretation is with respect to the `about()` condition as implemented by the retrieval model, and the structural constraint as implemented by the query processing engine, used by the XML information retrieval system. Sections "Scoring Strategies" and "Combination Strategies" describe approaches used to implement the `about` conditions, whereas section "Processing Structural Constraints" describes approaches used to process structural constraints, for the purpose of ranking XML elements for given queries.

Clause-Based Queries

Clause-based queries for XML information retrieval can be compared to SQL, the standard query language for (relational) databases. These queries are made of nested clauses to express information needs. The most prominent clause-based query languages for XML information retrieval are XQuery (http://www.w3.org/TR/xquery/) and XQuery Full-Text (http://www.w3.org/TR/xpath-full-text-10/).

XQuery is an XML query language that includes XPath as a sublanguage, but adds the possibility to query multiple documents and combine the results into new XML fragments. The core expressions of XQuery are the FLWOR expressions, which we illustrate with an example. The following query is a FLWOR expression that lists the authors, ordered by their last name, that have written at least 100 articles:

```
for $aut in (doc ("aut.xml")//author)
       let $c:=
              count (doc("article.xml")/article
                     [author=$aut])
                     where $c>100
                     order by $aut/lastname
                     return
                     <author> {$aut/lastname, $c} </author>
```

The `for` (F in FLWOR) clause binds the variable $aut so that it iterates over the author elements in the document "aut.xml" in the order that they appear. For every such binding, the `let` (L) clause binds the variable $c to the number of articles from author $aut (from the document "article.xml"). Those author elements for which the condition in the `where` (W) clause is true are selected, i.e., number of articles is above 100. The resulting bindings are sorted by the `order by` (O) clause on the author last name. Finally, the `return` (R) clause creates for each binding $aut and $c in the result of the preceding clause a new author element that contains the last name element of the author, and the associated number of articles.

XQuery is a powerful XML query language. However, its text search capabilities are limited and, in addition, the result is a set of (new) XML fragments; no ranking is performed. Thus its usefulness in XML information retrieval is limited. This has led to the development of XQuery Full-Text.[19]

XQuery Full-Text has been inspired by earlier query languages for searching structured text, e.g., ELIXIR,[20] JuruXML,[21] XIRQL.[17] The added text search capabilities come with the introduction of the *FTContainsExpr* expression, as shown in the following example:

© 2012 Taylor & Francis Group, LLC

```
//article[./title ftcontains {"XML", "retrieval"}
       all] // author
```

which returns the authors of articles whose title contains the words "XML" and "retrieval." XQuery Full-Text defines primitives for searching text, such as phrase, word order, word proximity, etc. For example, the following XQuery Full-Text expression:

```
//article[./title ftcontains {"XML", "retrieval"}
all window 6 words]//author
```

restricts the proximity of the matched words to appear within a window of six words in title elements.

To support the ranking of results, *FTScoreClause* expressions have been introduced to allow for the specification of score variables. For instance, the following query:

```
for $b score $s in //article [.//section ftcontains
       {"XML", "retrieval"} all]
              order by $s descending
              return <article title= "{$article/title}"
       score= "{$s}"/>
```

iterates over all articles whose sections contain both "XML" and "retrieval", where the $b variable binds the score of each such article to the score variable $s. These variables are used to return the titles of the articles and their scores in order of decreasing relevance.

XQuery Full-Text does not implement a specific scoring method, but it allows an implementation to proceed as it wishes. In other words, each XQuery Full-Text implementation can use a scoring method of its choice. Therefore, an appropriate implementation of XQuery Full-Text can allow ranking of results. From a user perspective, XQuery Full-Text may be viewed as far too complex, which is one of the reasons the INEX community developed NEXI, a path-based query language with less expressiveness than a clause-based query language, as its query language. A second one was to keep the construction of the test collections manageable, for instance during the assessment task (see explanation on "assessors" earlier in the entry). Nevertheless, XQuery Full-Text is needed in applications involving expert users, e.g., medical domain, patent industry, law.

REPRESENTATION STRATEGIES

To retrieve documents relevant to a query, the first task of an information retrieval system is to index all documents forming the searched collection. The indexing task aims to obtain a representation of the content of documents (i.e., what each document is about), which can then be used to score each document according to how relevant it is to a given query. Classical indexing strategies in information retrieval make use of term statistics, the most common ones being the within-document term frequency, *tf*, and the inverse document frequency, *idf*. *tf* is the number of occurrences of a term in a document and reflects how well a term captures the topic of a document; a term that occurs frequently in a document can be considered a good description of the document content (apart from common words, referred to as stop words, e.g., "the," "every" in the English language). *idf* is the inverse number of documents in which a term appears and is used to reflect how well a term discriminates between relevant and non-relevant documents; a term that appears in all documents of the collection is not good at discriminating between the content of two documents, and hence their relevance or nonrelevance.

With these term statistics, an index is build, for instance in the form of an inverted file, which gives for each term in the collection its *idf*, and for each document containing that term, the corresponding *tf*. Indexing algorithms for XML information retrieval require similar terms statistics, but at element level, i.e., they require so-called within-element term frequency, *etf*, and inverse element

© 2012 Taylor & Francis Group, LLC

frequency, *ief*. The indexing of a collection of documents involves other steps than calculating term statistics. These include tokenization, stop word removal, stemming, etc.[22] In XML information retrieval, the same steps are applied, and other steps such as parsing the XML format, which are not discussed in this entry. Also not discussed in this entry is that an index of the structure is build in order to record the relationships between elements.

In XML information retrieval, there are no a priori fixed retrieval units. The whole document, a part of it (e.g., one of its sections), or a part of a part (e.g., a paragraph in the section), that is, elements at all levels of granularity, all constitute potential answers to a given query. The simplest approach to allow the retrieval of elements at any level of granularity is to index all elements. Each element thus corresponds to a document, and conventional information retrieval representation techniques can be used. Term statistics (*etf* and *ief*) for each element are then calculated exactly in the same way as for *tf* and *idf* but based on the concatenation of the text of the element and that of its descendants.[23]

This is the most common approach. It however raises an issue because of the nested nature of the units forming an XML document: the *ief* value of a term will consider both the element that contains that term and all elements that do so in virtue of being ancestors of that element. For instance, for a section element composed of two paragraph elements, the fact that a term appears in the paragraph implies that it also appears in the section. This "double" occurrence may have an adverse effect with respect to using *ief* to discriminate between relevant and nonrelevant elements.

As a consequence, alternative means have been used to calculate *ief*. For instance, *ief* has been estimated across elements of the same type[24] or across documents.[25] The former greatly reduces the impact of nested elements on the *ief* value of a term, but does not eliminate it if elements of the same type are nested within each other (as it is the case with the Wikipedia test collection used at INEX[26]). The latter is the same as using inverse document frequency, which completely eliminates nested elements. Experimental results[27] indicate that estimating *ief* across documents shows slight improvement over using elements. However, other experimental results[28] show that better performance was obtained estimating *ief* across all elements than across elements of the same types. As of today, it is not yet clear what is the best way to estimate *ief*, whether the estimation strategy depends on the retrieval model and its artifacts used to rank elements, or whether the issue of nested elements actually matters. Further research is needed here.

An alternative to using the concatenated text in an element to estimate term statistics is to derive them through the aggregation of term statistics (both *etf* and *ief*) of the element's own text, and those of each of its children elements.[29,30] Aggregated-based ranking, discussed in section "Aggregation," uses the aggregated representation of elements to rank elements.

A second alternative approach is to only index leaf elements. A leaf element is one at the bottom of the document tree structure, i.e., an element with no children elements, or an element that is considered the smallest possible unit of retrieval. This implies that term statistics will only be calculated for leaf elements, which can then be used to rank the leaf elements themselves. With such strategy, the ranking of non-leaf elements requires propagation mechanisms (discussed in section "Propagation") that combine the score of their children elements into that of the element.[31] Both this and the above (aggregation) strategies overcome the issue of nested elements with respect to the calculation of *ief*.

It has also been common to discard elements smaller than a given threshold (usually expressed in terms of number of words),[23] which are often considered not meaningful retrieval units (they are too small to make much sense as results). It was however argued[32] that although the small elements should not be returned, they might still influence the scoring of enclosing elements, so they should still be indexed, in particular when a propagation mechanism for scoring non-leaf elements is used.

A final strategy,[25,33] referred to as selective indexing, is to only index those element types with the highest distribution of relevant elements in past relevance data. With this strategy, a separate index is built for each selected element type (e.g., for a collection of scientific articles, these types may include article, abstract, section, subsection, and paragraph). The statistics for each index are then calculated separately. Since each index is composed of terms contained in elements of the same

© 2012 Taylor & Francis Group, LLC

type (and likely comparable size), more appropriate term statistics are generated. In addition, this approach greatly reduces the term statistics issue arising from nested elements, although it may not eliminate it. At retrieval time, the query is then run in parallel on each index, and the list results (one for each index) are merged to provide a single list of results, as discussed in section "Merging."

It is not yet clear which indexing strategy is the best, as obviously which approach to follow would depend on the collection, the types of elements (i.e., the DTD), and their relationships. In addition, the choice of the indexing strategy has an effect on the ranking strategy. An interesting research would be to investigate all indexing strategies within a uniform and controllable environment to determine those leading to the best performance, across, or depending, on the ranking strategies.

RANKING STRATEGIES

Given an indexed collection of XML documents, the next task of an XML information retrieval system is to return for each submitted query, with or without structural constraints, a list of XML elements ranked in order of their estimated relevance to that query. In information retrieval, retrieval models are used to calculate what is called a retrieval score (usually a value between 0 and 1), which is then used as a basis to rank documents. Many of the retrieval models developed for unstructured text (document) retrieval have been adapted to XML information retrieval to provide such a score at element level (section "Scoring Strategies"). These scores may be used to directly generate the ranked list of elements, or as input to combination strategies required for some indexing strategies in order to rank elements at all levels of granularity (section "Combination Strategies"). For content-and-structure queries, in the context of INEX as expressed by the path-based query language NEXI (see section "Path-Based Queries"), the structural constraints must be processed to provide results that not only satisfy the content, but also the structural criteria of such queries (section "Processing Structural Constraints"). Finally, not all relevant elements should be returned as results, as they may contain overlapping content. This is because of the nested nature of XML documents, which often means that a parent and its child element may both be estimated as relevant, although to a different extent. Some processing is needed to deal with overlapping elements in result lists (section "Removing Overlaps").

Scoring Strategies

Whatever the representation strategy, i.e., whether all elements or only a subset of them are indexed, a scoring function is used to estimate the relevance of these elements for a given query. With the propagation strategy (discussed in section "Propagation"), the scoring function is applied to leaf elements only, whereas in other cases, it is applied to all potentially retrievable elements. Scoring functions used in XML information retrieval have been based on standard information retrieval models, such as the vector space, BM25, language models, to name a few. These have been adapted to incorporate XML-specific features. As an illustration, we describe a scoring function defined upon a language modeling framework inspired by Sigurbjornsson, et al.[23]

Given a query $q = (t_1, t_2, \ldots, t_n)$ made of n terms t_i, given an element e and its corresponding element language model θ_e, the scoring function expressed by $P(e|q)$ is defined as follow:

$$P(e|q) \propto P(e)P(q|\theta_e)$$

$P(e)$ is the prior probability of relevance for element e and $P(q|\theta_e)$ is the probability of a query being generated by the element language model θ_e, and can be calculated as

$$P(t_1, \ldots, t_n \mid \theta_e) = \prod_{i=1}^{n} \lambda P(t_i \mid e) + (1 - \lambda) P(t_i \mid C)$$

© 2012 Taylor & Francis Group, LLC

$P(t_i|e)$ is the probability of term t_i in element e, $P(t_i|C)$ is the probability of query term t_i in the collection, and λ is the smoothing parameter. $P(t_i|e)$ is the element model based on element term frequency (modeling *itf*), whereas $P(t_i|C)$ is the collection model, for example, based on inverse element frequency (modeling *ief*).

One important XML feature is the length of an element. Indeed, it was shown[34] that considering element length is necessary in XML information retrieval to cater for the wide range in element sizes. This can be incorporated by setting the prior probability $P(e)$ as follows:

$$P(e) = \frac{\text{length}(e)}{\sum_C \text{length}(e)}$$

where length(e) is the length of element e. Examples of other XML-specific features used in XML information retrieval include the path length,[35] the type of an element (its tag),[36] and the number of topics discussed in an element.[37]

The size of elements forming XML documents varies greatly. For example, compare a paragraph to a section in a 10-page scientific article. There are likely to be fewer terms indexing the paragraph than the section, leading to a higher chance of a vocabulary mismatch between a paragraph (or any small elements) and a query than between a section (or any large elements) and the same query. In addition, the fact that a paragraph element does not contain all query terms, but is contained in a section element that contains all query terms, is likely to be more relevant than if contained in a section element that does not contain all query terms.

More generally, the context of an element, i.e., the parent, all or some of its ancestors, or the entire document, can provide more evidence on what an element is or is not about. To incorporate the selected context(s) in estimating relevance, the score of an element is modified to include that of its (selected) context(s). The most common technique is to use the document containing the element as context (the document is also an element, albeit a large one, and corresponds to what is being referred to as the root element). This means combining the score of the element to that of the XML document containing that element, where the element and the document retrieval scores are estimated by an XML information retrieval model. The combination can be as simple as the average of the two scores.[38] A scaling factor can be used to emphasize the importance of one score compared to the other.[33] This technique (using element and document scores) has been shown to increase retrieval performance, in particular for long documents, and has been widely used in XML information retrieval.

Combination Strategies

Three of the representation strategies described in section "Representation Strategies" require combination strategies to provide a rank list of all potentially retrievable elements. These combination strategies are propagation (section "Propagation"), aggregation (section "Aggregation"), and merging (section "Merging").

Propagation

The propagation strategy is needed with the representation strategy that only indexes leaf elements. The relevance of the leaf elements for given queries is estimated on this indexing, resulting in retrieval scores for leaf elements. The relevance of non-leaf elements is estimated through a propagation mechanism, where the retrieval score of a non-leaf element is calculated on the basis of the retrieval scores of its descendant elements. The propagation starts from the leaf elements and moves upward in the document tree structure.

© 2012 Taylor & Francis Group, LLC

The most common propagation mechanism consists of a weighted sum of retrieval scores. For instance, the number of children elements of an element has been used as a weight[31]:

$$\text{score}(e,q) = D(m)\sum_{e_c}\text{score}(e_c q)$$

where score(e,q) is the retrieval score of an element with respect to query q, e_c is a child element of e, m is the number of retrieved children elements of e, $D(m) = 0.49$ if $m = 1$ (e has only one retrieved child element), and 0.99 otherwise. The value of $D(m)$, called the decay factor, depends on the number of retrieved children elements. If e has one retrieved child then the decay factor of 0.49 means that an element with only one retrieved child will be ranked lower than its child. If e has several retrieved children, the decay factor of 0.99 means that an element with many retrieved children will be ranked higher than its children elements. Thus, a section with a single relevant paragraph would be considered less relevant than the paragraph itself, as it is simply better to return the paragraph as returning the section does not add anything more. On the other hand, a section with several retrieved paragraphs will be ranked higher than any of the paragraphs, as it will allow users to access these several paragraphs through the returned section.

This approach, known as the GPX model, has been very successful within INEX, across test collections and retrieval scenarios. Another successful approach, implemented in the XFIRM system,[32] is to define the weight used in the propagation based on the distance between an element and its retrieved leaf elements.

Aggregation

This combination strategy is applied when the representation of an XML element is defined as the aggregation of the representation of its own content (if any) and the representations of the content of its children elements (if any). Retrieval is then based on these aggregated representations. The representation of the element's own content is generated using standard indexing techniques, whereas an aggregation function is used to generate the representation of the non-leaf elements. The aggregation function can include parameters (referred to as, e.g., augmentation factor[29]) specifying how the representation of an element is influenced by that of its children elements (a measure of the contribution of, for instance, a section to its embedding chapter). Aggregation is to be contrasted to propagation; in the former, the combination is applied to representations, whereas in the latter, it is applied to retrieval scores.

To illustrate aggregation, we describe an approach based on the language modeling framework inspired from Ogilvie and Callan.[30] There, each element e is modeled by a language model θ_{eown} based on its own content. Now assume that e has several children, e_j, each with their own language model θ_{ej}. Let $P(t|\theta_{\text{eown}})$ and $P(t|\theta_{ej})$ be the probability of query term t being generated by the language models θ_{eown} and θ_{ej}, respectively. The language model, called θ_e, modeling the element e based on its own content and that of its children, is defined as a linear interpolation of language models:

$$P(t \mid \theta_e) = \lambda_{\text{own}} P(t \mid \theta_{e_{\text{own}}}) + \sum_{e_j} \lambda_j P(t \mid \theta_{e_j})$$

where

$$\lambda_{\text{own}} + \sum_{e_j} \lambda_j = 1$$

© 2012 Taylor & Francis Group, LLC

The λ parameters model the contribution of each language model (i.e., element) in the aggregation, here implemented as a linear combination. The ranking of the elements is then produced by estimating the probability that each element generates the query (e.g., similarly to the formulation described in section "Scoring Strategies"). The effectiveness of the aggregation, however, depends heavily on the appropriate settings of the λ parameters, whose values are usually estimated through learning methods.

Merging

The last combination strategy is that of merging, which is needed when a selective indexing strategy is used. With this indexing strategy, a separate index is created for each selected type of elements (e.g., article, abstract, section, paragraph, etc.). A query submitted to the XML information retrieval system is run against each index separately, resulting in separate ranked lists of, e.g., article elements, section elements, paragraph elements, etc. These lists need to be merged to provide a single ranking, across all element types.

In Mass and Mandelbrod,[33] the vector space model is used to rank elements in each index. Let e be an element and q a query. The following scoring function is used:

$$\text{score}(e,q) = \frac{\sum_{t \in q} w(t,q) \times w(t,e) \times ief(t)}{\|q\| \times \|e\|}$$

where $w(t,q)$ is the term weight based on within-element (etf)/query term frequency, and $ief(t)$ is the inverse element frequency. To merge the lists, normalization is performed to take into account the variation in size of the elements in the different indices (e.g., paragraph index vs. article index). For each result list, the element scores are normalized with score(q, q), which corresponds to the score of the query as if it was an element in the collection run against the corresponding index. This ensures that scores across indices are comparable. The lists are then merged based on the normalized scores.

Processing Structural Constraints

We described so far approaches that were developed and evaluated during the INEX campaigns to rank elements given the content condition of a query. Given a query consisting of terms, these approaches deliver a list of elements ranked according to how they have been estimated relevant to that query. As discussed in section "Query Languages," content-and-structure query languages have been developed to allow users to specify structural constraints, e.g., "give me a section in an article about XML technology that also discusses in one of its section book search."

Within INEX, structural constraints are viewed as hints as to where to look to find relevant information. The reasons for this view are twofold. First, it is well-known that users of information retrieval systems do not always, or simply cannot, properly express the content criterion (i.e., select the most useful query terms) of their information need. It is very likely that this also holds for the structural criterion of the information need. For instance, a user asking for paragraph components on "XML retrieval evaluation measures" may not have realized that relevant content for this query is scattered across several paragraphs, all of them contained within a single section. For that user, it may make more sense to return the whole section instead of individual paragraphs. Second—and to some extent as a consequence of the first reason above—there is a strong belief in the XML information retrieval community that satisfying the content criterion is, in general, more important than satisfying the structural criterion. For instance, even if a user is looking for section components on a particular topic, returning to that user abstract components would still be satisfactory, as long as the content criterion is satisfied.

© 2012 Taylor & Francis Group, LLC

Two main approaches have been developed to process structural constraints in XML information retrieval following this so-called vague interpretation of the structural constraints in content-and-structure queries. A first approach is to build a dictionary of equivalent synonyms. If, for example, <p> corresponds to paragraph type and <p1> corresponds to the first paragraph in a sequence of paragraphs, it would be quite logical to consider <p> and <p1> as equivalent tags.[39,40] The dictionary can also be built from analyzing past relevance data.[41] If in such a data set, for example, a query asked for <section> elements, then all types of elements assessed relevant for that query can be considered equivalent to the <section> tag. Thus with this approach, if the structural constraint refers to, e.g., <section>, then any element that is of type considered equivalent to <section>, will satisfy that structural constraint.

A second technique is that of structure boosting. There, the retrieval score of an element is generated ignoring the structural constraint of the query, but is then boosted according to how the structural constraint is satisfied by the element. The element structure and the query structure are compared and a structure score is generated. This structure score can be based on comparing the paths,[21,24] and/or the tags in the paths.[42] An important issue here is to determine the appropriate level of boosting, i.e., how much the initial content-based score should be boosted by the structure score.

The above techniques and their variants are mostly used to determine the relevance of an element according to the content condition and tag-based like structural constraints, e.g., "retrieve sections about XML retrieval." For more complex structural constraints, as allowed by a path-based language such as NEXI, e.g., "retrieve paragraphs about ranking algorithms contained in sections about XML retrieval," a first step is usually applied, which is to divide the query into two tag-based like subqueries, e.g., "retrieve paragraphs about ranking algorithms" and "retrieve sections about XML retrieval." Each subquery is then processed according to its content condition, and its tag-based like structural condition as described in the previous two paragraphs. Each subquery results in a ranked list of elements. To generate a ranked list for the whole query, the two ranked lists are compared, e.g., only elements returned for the "paragraph" subquery whose ancestors are contained among the elements returned for the "section" subquery are then retrieved. The final score depends on the implementation of the contain operation, e.g., a simple set containment, or using fuzzy operators.[43]

Techniques for processing structural constraints were evaluated in the context of INEX, where the relevance of an element was assessed based on content only. In other words, there was no assessment of whether, for instance, a section element was a better element type to return than another element type (if both were relevant according to their content). Also, considering the structural constraints did not usually increase retrieval performance, apart maybe at very early ranks.[44] This result may however be due to the evaluation methodology. More research is needed regarding the usefulness and the impact of structural constraints for XML information retrieval.

Removing Overlaps

We recall that the aim of an XML information retrieval system is to return the most relevant elements for a given query. Because of the nested structure of XML documents, when an element has been estimated relevant to a given query (by any of the XML ranking strategies presented in this entry), it is likely that its ancestors will also be estimated as relevant, although likely to a different extent. This is because the same text fragment can be contained in several elements along a same path (e.g., a paragraph, its enclosing subsection, the enclosing section, etc). Thus the element itself, its ancestors, and a number of its descendants may be contained in the result list, eventually leading to a considerable amount of redundant information being returned to users, which may not be acceptable to them.[45]

The outcome of any of the ranking strategies described so far in section "Ranking Strategies" is a list of elements ranked according to their estimated relevance to a given query, without looking at

© 2012 Taylor & Francis Group, LLC

the overlap issue. XML information retrieval systems may have to decide which elements should be returned from a list of relevant but overlapping elements. Several approaches have been proposed to generate overlap-free result lists. Their starting point usually consists of the list of elements returned as results to a query, which they then process.

The most common approach, referred to as brute-force filtering, selects the highest ranked element from the result list and removes any ancestor and descendent elements from lower ranks. The process is applied recursively. This approach relies on the provision of ranking strategies that rank, among overlapping elements, those that should be selected at higher ranks. However, the ranking may not be appropriate for the purpose of returning the list of the most relevant non-overlapping results. This has led to a number of alternative approaches, where the tree structure of the XML documents has been considered to decide which elements to remove from a list of overlapping results.

In the first such approach,[41] a notion of the usefulness of an element is introduced to decide which elements to remove. Usefulness is modeled through a utility function defined upon the retrieval score of an element, its size, and the amount of irrelevant information contained in its children elements (implemented as the "amount" of text contained in the non-retrieved children elements). An element with an estimated utility value higher than the sum of the utility values of its children is selected and its children are removed. Otherwise, the children elements whose utility values exceed some set threshold are selected and the element is removed.

An alternative approach[46] looks at the distribution of retrieved elements in the XML document's tree structure, in addition to their score, to select the elements to retain. For instance, an element that has many of its descendants retrieved, but which are evenly distributed in the corresponding subtree structure, and in addition has a similar score to the parent element, is selected. This is because already from that selected element, all its descendants, many of which are being estimated as relevant, can be accessed. Otherwise, its descendants are selected to be themselves further processed.

A third approach[47] calculates a new score for each element on the basis of the retrieval scores of its (if any) descendent elements. This is done through a bottom-up propagation mechanism, using for instance the maximum or average operation to recalculate the scores. These scores are used to generate a new ranked list, which is then filtered by selecting the highest ranked elements, and then removing either all ancestors or all descendants of that selected element from the list (e.g., brute-force filtering). The best performances were obtained using the maximum function and removing the descendants.

Techniques that explicitly considered the document logical (tree) structure to remove overlaps usually outperformed those that did not. There is, however, the issue of speed, as the removal of overlaps is done at query time, thus requiring efficient implementations. An interesting question would be to investigate the effect of the original result list (how good it is, and how we define "goodness") on the overlap removal strategy. There are indications that a good initial result list, where good depends on the definition of relevance in the context of XML information retrieval, leads to better overlap-free result list than a less good one.[48]

DISCUSSION

XML information retrieval research is related to work on structured document retrieval. The term "structured document retrieval," which was introduced by the information retrieval community, refers to "passage retrieval" and "structured text retrieval." In passage retrieval, documents are first decomposed into passages (e.g., fixed-size text-windows of words,[49] fixed discourses such as paragraphs,[4] or topic segments through the application of a topic segmentation algorithm[50]). Passages are then retrieved as answers to a query (and have also been used to rank documents as answers to the query). Since 2007, INEX has a passage retrieval task.[51]

Structured text retrieval is concerned with the development of models for querying and retrieving from structured text,[52] where the structure is usually encoded with the use of mark-up languages,

© 2012 Taylor & Francis Group, LLC

such as SGML, and now predominantly XML. Examples of pre-XML/NEX approaches include.[1,3,53] Most of the early structured text retrieval models, however, do not return a ranked list of results. Approaches that were specifically developed for XML retrieval, but still pre-INEX include.[54–57] A survey on indexing and searching XML retrieval is from Luk, et al.,[58] and two workshops on XML retrieval held at the SIGIR (http://www.sigir.org/) conference were reported.[5,7] A recent overview on XML retrieval research is from Amer-Yahia and Lalmas.[59]

Research on XML retrieval significantly flourished since the set-up of INEX, as the latter allowed the evaluation and comparison of XML information retrieval approaches. Nowadays, XML retrieval is almost a synonym for structured document retrieval, or structured text retrieval. In this entry, we described many of the strategies used for representing and ranking XML elements, which were experimented on the INEX test collections. We also described query languages that were developed to access XML documents with respect to both content and structural conditions.

It is not yet possible to state which approaches, whether for querying, representing, or ranking, or their combination, work best, since many factors are involved when deciding how relevant an element is to a given query (e.g., the size of an element, the type of element, the relevance of structurally related elements, the interpretation of the structural constraint, etc.). Indeed, XML information retrieval can be regarded as a combination problem, where the challenge is to decide which evidence to combine and how to combine it for effective retrieval. We can however postulate that considering the context of the element, the size of the element, and the element's own content (directly or using a propagation or aggregation strategy) to estimate that element relevance to a given query has been shown to be beneficial for XML information retrieval. An open research question is the processing of structural constraints, as so far, only limited improvement in retrieval performance has been observed with structure-and-content queries.

The querying, representation, and ranking strategies described in this entry have been developed with the purpose of estimating the relevance of an element for a given query, which is only one retrieval scenario. This may not necessarily be the end task in XML information retrieval. Returning overlap-free results is another retrieval scenario, one where users do not want redundant information (approaches developed for this purpose were described in section "Removing Overlaps"). Another retrieval scenario investigated at INEX is the relevant in context task.[60] This task is concerned with returning the most relevant documents for a given query, and within each document, identifying the most relevant elements. Such a retrieval scenario was identified as important, if not expected, in a user study carried out within a software company regarding the benefit of focused retrieval.[61]

In the relevant in context task, elements from the same documents are grouped together. A system could also create so-called "fictitious" documents, i.e., new documents made from some intelligent aggregation of elements coming from different documents, which is another retrieval scenario, currently receiving increasing attention in information retrieval research.[62] XQuery Full-Text would be an appropriate language for this task, as it allows the specification of new XML fragments to return as results.

Another retrieval scenario, also investigated at INEX, is the best in context task.[60] There, the aim is to identify the one and only best entry point in a document, i.e., the XML element, from where one could start reading relevant content. Such a retrieval scenario makes sense with a collection of relatively medium size documents (e.g., Wikipedia documents used at INEX since 2006[26]).

Although not discussed in this entry, two important issues in XML information retrieval are interface and interaction. Appropriate interfaces are needed to cater for the richer and likely more complex interaction between users and XML information retrieval systems, for example, with respect to expressing content-and-structure queries.[63] Since 2004, INEX runs an interactive track (iTrack) that looked at interaction issues in XML information retrieval.[64] One outcome of iTrack is that users did not like being returned (at least too much) redundant information (overlapping results). This led to the development of algorithms specifically dedicated to remove or reduce overlaps (see section "Removing Overlaps"). A second outcome was that users expected to have not only access

© 2012 Taylor & Francis Group, LLC

to relevant elements, but also the context of these elements (e.g., the document containing, or the parent element of, a retrieved element). This led to the proposal of a table of a content shown in conjunction to the element being accessed[65] or the use of heatmap highlighting relevant elements within a retrieved document.[66]

Information retrieval approaches developed for querying, representing, or ranking are relevant to applications concerned with the effective access to repositories of documents annotated in XML, or similar mark-up languages. XML retrieval is becoming increasingly important in all areas of information retrieval, and in particular in the area of so-called focused retrieval.[67] Current applications of XML information retrieval technologies already exist.[68] An example is that of book search,[69] which is a research track being investigated at INEX since 2007.[70]

ACKNOWLEDGMENTS

This entry is based on two other entries on XML information retrieval cowritten by the author, a book chapter on "Structured Text Retrieval" to appear in the second edition of Baeza-Yates and Ribeiro-Neto,[71] and an entry on "XML Retrieval"[72] to appear in the Encyclopedia of Database Systems.[73] The author would like to thank Benjamin Piwowarski and Anastasio Tombros for their comments on this entry.

REFERENCES

1. Burkowski, F. Retrieval activities in a database consisting of heterogeneous collections of structured text. In 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Copenhagen, Denmark, 1992; 112–125.
2. Clarke, C.A.; Cormack, G.; Burkowski, F. An algebra for structured text search and a framework for its implementation. Comput. J. **1995**, *38* (1), 43–56.
3. Navarro, G.; Baeza-Yates, R. Proximal nodes: A model to query document databases by content and structure. ACM Trans. Inform. Syst. **1997**, *15* (4), 400–435.
4. Wilkinson, R. Effective retrieval of structured documents. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*; Springer-Verlag: New York, Inc., 1994; 311–317.
5. Baeza-Yates, R.; Fuhr, N.; Maarek, Y. Second edition of the "XML and information retrieval" workshop held at SIGIR' 2002, Tampere, Finland. SIGIR Forum **2002**, *36* (2), 53–57.
6. Blanken, H.; Grabs, T.; Schek, H.-J.; Schenkel, R.; Weikum, G., Eds. *Intelligent Search on XML Data, Applications, Languages, Models, Implementations, and Benchmarks*; Springer: New York, 2003; Vol. 2818.
7. Carmel, D.; Maarek, Y.S.; Soffer, A. XML and information retrieval: A SIGIR 2000 workshop. SIGIR Forum **2000**, *34* (1), 31–36.
8. Gövert, N.; Kazai, G. Overview of the initiative for the evaluation of XML retrieval (INEX) 2002. In First Workshop of the INitiative for the Evaluation of XML Retrieval (INEX), Schloss Dagstuhl, Germany; 2002; 1–17.
9. Fuhr, N.; Go¨vert, N.; Kazai, G.; Lalmas, M., Eds. *INitiative for the Evaluation of XML Retrieval (INEX)*. In Proceedings of the First INEX Workshop. Dagstuhl, Germany, December 8–11, 2002, Sophia Antipolis: France, 2003. ERCIM Workshop Proceedings, ERCIM.
10. Fuhr, N.; Kamps, J.; Lalmas, M.; Malik, S.; Trotman, A., Eds. Focused Access to XML Documents, 6th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2007, Dagstuhl Castle, Germany, December 17–19, 2007, Selected papers, 2008.
11. Fuhr, N.; Lalmas, M.; Malik, S., Eds. INitiative for the evaluation of XML retrieval (INEX). In *Proceedings of the Second INEX Workshop*, Dagstuhl, Germany, December 15–17, 2003, 2004.
12. Fuhr, N.; Lalmas, M.; Malik, S.; Kazai G., Eds. Advances in XML information retrieval and evaluation. In Fourth Workshop of the INitiative for the Evaluation of XML Retrieval (INEX 2005). 2006; Vol. 3977 of Lecture Notes in Computer Science Springer-Verlag.
13. Fuhr, N.; Lalmas, M.; Malik, S.; Szlavik, Z., Eds. Advances in XML information retrieval. In Third International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2004, Dagstuhl Castle, Germany, December 6–8, 2004, Revised selected papers, 2005; Vol. 3493 of Lecture Notes in Computer Science, Springer.

© 2012 Taylor & Francis Group, LLC

14. Fuhr, N.; Lalmas, M.; Trotman, A., Eds. Comparative evaluation of XML information retrieval systems. In 5th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2006, 2007; Vol. 4518 of Lecture Notes in Computer Science, Springer-Verlag.
15. Cohen, S.; Mamou, J.; Kanza, Y.; Sagiv, Y. XSEarch: A semantic search engine for XML. In 29th International Conference on Very Large Data Bases, Berlin, Germany; 2003; 45–56.
16. Theobald, A.; Weikum, G. The index-based XXL search engine for querying XML data with relevance ranking. In EDBT, 2002; Springer-Verlag: London; 477–495.
17. Fuhr, N.; Grossjohann, K. XIRQL: An XML query language based on information retrieval concepts. ACM Trans. Inform. Syst. **2004**, *22* (2), 313–356.
18. Trotman, A.; Sigurbjornsson, B. Narrowed extended Xpath I (NEXI). In Advances in XML Information Retrieval, Third International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2004, Dagstuhl Castle, Germany, Revised selected papers, 2005; 16–40.
19. Amer-Yahia, S.; Botev, C.; Dörre, J.; Shanmugasundaram, J. Full-text extensions explained. IBM Syst. J. **2006**, *45* (2), 335–352.
20. Chinenyanga, T.T.; Kushmerick, N. Expressive retrieval from XML documents. In 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, New Orleans, LA, 2001; 163–171.
21. Carmel, D.; Maarek, Y.; Mandelbrod, M.; Mass, Y.; Soffer, A. Searching XML documents via XML fragments. In 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Toronto, ON, Canada, 2003; 151–158.
22. Manning, C.; Raghavan, P.; Schutze, H., Eds. *Introduction to Information Retrieval*; Cambridge University Press: Cambridge, 2008.
23. Sigurbjornsson, B.; Kamps, J.; de Rijke, M. An element-based approach to XML retrieval. In Proceedings INEX 2003 Workshop, Schloss Dagstuhl, Germany, 2004; 19–26.
24. Theobald, M.; Schenkel, R.; Weikum, G. TopX and XXL at INEX 2005. In Advances in XML Information Retrieval and Evaluation, 4th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2005, Dagstuhl Castle, Germany, Revised selected papers, 2006; 282–295.
25. Clarke, C. Controlling overlap in content-oriented XML retrieval. In 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Salvador, Brazil, 2005; 314–321.
26. Denoyer, L.; Gallinari, P. The Wikipedia XML corpus. SIGIR Forum **2006**, *40* (1), 64–69.
27. Ramírez, G. Structural features in XML retrieval, Ph.D. thesis, University of Amsterdam, Amsterdam, 2007.
28. Broschart, A.; Schenkel, R.; Theobald, M.; Weikum, G. TopX @ INEX 2007. In Focused access to XML documents, 6th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2007, Dagstuhl Castle, Germany, Selected papers, 2008.
29. Gövert, N.; Abolhassani, M.; Fuhr, N.; Grossjohann, K. Content-oriented XML retrieval with HyRex. In First Workshop of the INitiative for the Evaluation of XML Retrieval (INEX), Schloss Dagstuhl, Germany, 2002; 26–32.
30. Ogilvie, P.; Callan, J. Hierarchical language models for XML component retrieval. In Advances in XML Information Retrieval, Third International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2004, Dagstuhl Castle, Germany, Revised Selected Papers, 2005; 224–237.
31. Geva, S. GPX—gardens point XML IR at INEX 2005. In Advances in XML Information Retrieval and Evaluation, 4th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2005, Dagstuhl Castle, Germany, Revised selected papers, 2006; 240–253.
32. Sauvagnat, K.; Hlaoua, L.; Boughanem, M. XFIRM at INEX 2005: Ad-hoc and relevance feedback tracks. In Advances in XML Information Retrieval and Evaluation, 4th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2005, Dagstuhl Castle, Germany, Revised selected papers, 2006; 88–103.
33. Mass, Y.; Mandelbrod, M. Component ranking and automatic query refinement for XML retrieval. In Advances in XML Information Retrieval, Third International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2004, Dagstuhl Castle, Germany, Revised selected papers, 2005; 73–84.
34. Kamps, J.; de Rijke, M.; Sigurbjörnsson, B. Length normalization in XML retrieval. In 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Sheffield, U.K., 2004; 80–87.
35. Huang, F.; Watt, S.; Harper, D.; Clark, M. Compact representations in XML retrieval. In Comparative Evaluation of XML Information Retrieval Systems, 5th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2006, Dagstuhl Castle, Germany, Revised and selected papers, 2006; 64–72.

© 2012 Taylor & Francis Group, LLC

36. Gery, M.; Largeron, C.; Thollard, F. Probabilistic document model integrating XML structure. In INEX 2007 Pre-Proceedings, 2007; 139–149.
37. Ashoori, E.; Lalmas, M.; Tsikrika, T. Examining topic shifts in content-oriented XML retrieval. Int. J. Dig. Libr. **2007**, *8* (1), 39–60.
38. Arvola, P.; Junkkari, M.; Kekäläinen, J. Generalized contextualization method for XML information retrieval. In ACM CIKM International Conference on Information and Knowledge Management, Bremen, Germany, 2005; 20–27.
39. Mass, Y.; Mandelbrod, M. Retrieving the most relevant XML Components. In INEX 2003 Proceedings, 2003; 53–58.
40. Sauvagnat, K.; Boughanem, M.; Chrisment, C. Answering content and structure-based queries on XML documents using relevance propagation. Inform. Sys. **2006**, *31* (7), 621–635.
41. Mihajlovic, V.; Ramírez, G.; Westerveld, T.; Hiemstra, D.; Blok, H.E.; de Vries, A. TIJAH scratches INEX 2005: Vague element selection, image search, overlap, and relevance feedback. In Advances in XML Information Retrieval and Evaluation, 4th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2005, Dagstuhl Castle, Germany, Revised selected papers, 2006; 72–87.
42. van Zwol, R. B^3-SDR and effective use of structural hints. In Advances in XML Information Retrieval and Evaluation, 4th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2005, Dagstuhl Castle, Germany, Revised selected papers, 2006; 146–160.
43. Vittaut, J.-N.; Piwowarski, B.; Gallinari, P. An algebra for structured queries in Bayesian networks. In Advances in XML Information Retrieval, Third International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2004, Dagstuhl Castle, Germany, December 6–8, 2004; Revised selected papers, 2004; 100–112.
44. Trotman, A.; Lalmas, M. Why structural hints in queries do not help XML retrieval. In 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Seattle, WA, 2006; 711–712.
45. Tombros, A.; Malik, S.; Larsen, B. Report on the INEX 2004 interactive track. SIGIR Forum **2005**, *39* (1), 43–49.
46. Mass, Y.; Mandelbrod, M. Using the INEX environment as a test bed for various user models for XML retrieval. In Advances in XML Information Retrieval and Evaluation, 4th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2005, Dagstuhl Castle, Germany, Revised Selected Papers, 2006; 187–195.
47. Popovici, E.; Ménier, G.; Marteau, P.-F. SIRIUS XML IR system at INEX 2006: Approximate matching of structure and textual content. In Comparative Evaluation of XML Information Retrieval Systems, 5th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2006, Dagstuhl Castle, Germany, Revised and selected papers, 2007; 185–199.
48. Ashoori, E. Using topic shifts in content-oriented XML retrieval, Ph.D. thesis, University of London, Queen Mary, 2009.
49. Callan, J. Passage-level evidence in document retrieval. In Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval; Springer-Verlag: New York, Inc., 1994; 302–310.
50. Hearst, M. TextTiling: Segmenting text into multi-paragraph subtopic passages. Comput. Linguist. **1997**, *23* (1), 33–64.
51. Kamps, J.; Pehcevski, J.; Kazai, G.; Lalmas, M.; Robertson, S. INEX 2007 evaluation metrics. In Focused Access to XML Documents, 6th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2007, Dagstuhl Castle, Germany, Selected papers, 2008.
52. Baeza-Yates, R.A.; Navarro, G. Integrating contents and structure in text retrieval. SIGMOD Rec. **1996**, *25* (1), 67–79.
53. Macleod, I. Storage and retrieval of structured documents. Inform. Process. Manage. **1990**, *26* (2), 197–208.
54. Chiaramella, Y.; Mulhem, P.; Fourel, F. A model for multimedia information retrieval, Tech. rep, University of Glasgow, Glasgow, 1996.
55. Lalmas, M. Dempster-shafer's theory of evidence applied to structured documents: Modelling uncertainty. In 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Philadelphia, PA, 1997; 110–118.
56. Rölleke, T.; Lalmas, M.; Kazai, G.; Ruthven, I.; Quicker, S. The accessibility dimension for structured document retrieval. In Advances in Information Retrieval, 24th BCS-IRSG European Colloquium on IR Research, Glasgow, U.K., 2002; 284–302.
57. Schlieder, T.; Meuss, M. Result ranking for structured queries against xml documents. In DELOS Workshop: Information Seeking, Searching and Querying in Digital Libraries, Zurich, Switzerland, 2000.

© 2012 Taylor & Francis Group, LLC

58. Luk, R.P.; Leong, H.V.; Dillon, T.; Chan, A.S.; Croft, W.B.; Allan, J. A survey in indexing and searching XML documents. J. Am. Soc. Inform. Sci. Technol. **2002**, *53* (6), 415–437.
59. Amer-Yahia, S.; Lalmas, M. XML search: Languages, INEX and scoring. SIGMOD Rec. **2006**, *35* (4), 16–23.
60. Malik, S.; Trotman, A.; Lalmas, M.; Fuhr, N. Overview of INEX 2006. In Comparative Evaluation of XML Information Retrieval Systems, 5th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2006, Dagstuhl Castle, Germany, December 17–20, 2006; Revised and selected papers, 2007; 1–11.
61. Betsi, S.; Lalmas, M.; Tombros, A.; Tsikrika, T. User expectations from XML element retrieval. In 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Seattle, WA, 2006; 611–612.
62. Lalmas, M.; Murdock, V., Eds. ACM SIGIR Workshop on Aggregated Search. Singapore, 2008.
63. Zwol, R.; Baas, J.; van Oostendorp, H.; Wiering, F. Bricks: The building blocks to tackle query formulation in structured document retrieval. In Advances in Information Retrieval, 28th European Conference on IR Research, ECIR 2006, London, 2006; 314–325.
64. Tombros, A.; Larsen, B.; Malik, S. The interactive track at INEX 2004. In Advances in XML Information Retrieval, Third International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2004, Dagstuhl Castle, Germany, Revised selected papers, 2005; 410–423.
65. Szlávik, Z.; Tombros, A.; Lalmas, M. Feature- and query-based table of contents generation for xml documents. In Advances in Information Retrieval, 29th European Conference on IR Research, ECIR 2007, Rome, Italy, April 2–5, 2007; 456–467.
66. Kamps, J.; Koolen, M.; Sigurbjörnsson, B. Filtering and clustering XML retrieval results. In Comparative Evaluation of XML Information Retrieval Systems, 5th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2006, Dagstuhl Castle, Germany, Revised and Selected Papers, 2006; 121–136.
67. Trotman, A.; Geva, S.; Kamps, J. Report on the SIGIR 2007 workshop on focused retrieval. SIGIR Forum **2007**, *41* (2), 97–103.
68. Pharo, N.; Trotman, A. The use case track at INEX 2006. SIGIR Forum **2007**, *41* (1), 64–66.
69. Kantor, P.B.; Kazai, G.; Milic-raylying, N.; Wilkinson, R., Eds. Proceedings of the 2008 ACM Workshop on Research Advances in Large Digital Book Repositories, Books-Online 2008, Napa Valley, CA, October 30, 2008, 2008, ACM.
70. Kazai, G.; Doucet, A. Overview of the INEX 2007 book search track (BookSearch '07). SIGIR Forum **2008**, *42* (1), 2–15.
71. Baeza-Yates, R.A.; Ribeiro-Neto, B. *Modern Information Retrieval*; ACM Press/Addison-Wesley: New York, 1999.
72. Lalmas, M.; Trotman, A. XML retrieval. In *Encyclopedia of Database Systems*; Ozsu, M.; Liu, L., Eds. Springer, 2009.
73. Ozsu, M.; Liu, L., Eds. *Encyclopedia of Database Systems*; Springer, 2009.

© 2012 Taylor & Francis Group, LLC

31 Recommender Systems and Expert Locators

Derek L. Hansen, Tapan Khopkar, and Jun Zhang

CONTENTS

INTRODUCTION

Despite the abundance of recorded information, many information seekers turn to other humans for advice and recommendations. Humans, after all, can be quite adept at identifying and solving problems, summarizing relevant content, generating new ideas, and personalizing information. In addition, for some, interacting with other humans is far more socially enjoyable than interacting with static content. Thus, it is no surprise that the Internet is as much a platform for social interaction as it is a document repository.

System designers have taken advantage of the fact that so much social action is captured online by creating systems that extend traditional word-of-mouth exchanges. This entry discusses two such systems: *recommender systems* that provide personalized recommendations (e.g., movie suggestions) for items of potential interest, and *expert locator systems* that automatically identify experts on a particular topic of interest making it possible to obtain personalized advice from knowledgeable individuals outside of one's immediate social network. Expert locator systems can be thought of as a subset of recommender systems where experts are the "items" being recommended. We treat them separately in this entry because in practice there are often important distinctions between

© 2012 Taylor & Francis Group, LLC

recommending items and people. The remainder of the entry defines these two types of systems, outlines their key characteristics, provides some historical and current examples, and identifies the key research questions related to them. It concludes with a discussion about the importance and potential of these techniques given the increased amount of activity that can be digitally captured.

RECOMMENDER SYSTEMS

People are often confronted with situations where they need to assess the potential value of something that they have never experienced before. We need to find a new book to read, choose a doctor, and know which business is credible. When confronted with these situations, we often turn to experts or peers for recommendations. Increasingly, people receive recommendations from automated tools called recommender systems. For example, people browsing a book at Amazon are presented with a list of related books of potential interest. The related books (listed under the "People who bought this book also bought" header) are identified by a recommender system that relies upon the historical purchasing patterns of Amazon customers. More generally, recommender systems suggest items of potential interest to individuals who do not have personal experience with the items.

Though recommender systems have gained visibility with the spread of the Internet, they are not confined to the Web. Better Business Bureaus, Zagat's restaurant reviews, and The Times Book Review are some examples of recommender systems that predate the Web. Two things make Web-based recommender systems fundamentally different. First, they are able to provide personalized recommendations tailored to individuals. In contrast to The Times Book Review that provides the same recommendations to every reader, recommender systems can recommend a book based on the other books you personally enjoy, while not recommending it to others with different tastes. Second, recommender systems are able to base recommendations on the data from the masses, not just a handful of reviewers and editors. The online environment has also made it easier to efficiently capture people's preferences and easily distribute recommendations.

Broadly speaking, recommender systems can be classified into two types: collaborative and content-based. Collaborative recommender systems make recommendations based on the prior experience of other users, while content-based systems make recommendations based on features or descriptions of the items themselves. There are also hybrid recommender systems that mix these two approaches. For simplicity, we present them independently in the following sections.

Collaborative Systems

Collaborative recommender systems draw on the historical experience, or preferences of some users to make recommendations to other users. These systems are also called "Collaborative Filters" or "Social Filters" and have been used in a variety of settings to recommend newsgroup articles (e.g., GroupLens), books (e.g., Amazon's "People who bought this also bought" feature), movies (e.g., MovieLens), music (e.g., Last.fm), and even people (e.g., eBay's feedback system and other expert locators systems discussed in the following section).

Last.fm is a good example of a collaborative recommender system that offers many different recommendations. The site is a music portal that allows users to listen to music, find new music they are likely to enjoy, and find people with similar music tastes. When a user logs into the site, he can enter in his favorite songs and bands and add them to his "playlist," one of the components of his user profile. The system then compares the user's profile with the profiles of other Last.fm users. This comparison makes it possible to identify users who have similar tastes (e.g., people who like and dislike the same music). These individuals are shown as "neighbors" and a user is presented with opportunities to view their music lists. In this way, the system recommends people based on the similarity of user profiles. The system also provides a user with the option of hearing songs that his neighbors as a collective enjoy. Furthermore, when visiting an artist's home page a user can have

© 2012 Taylor & Francis Group, LLC

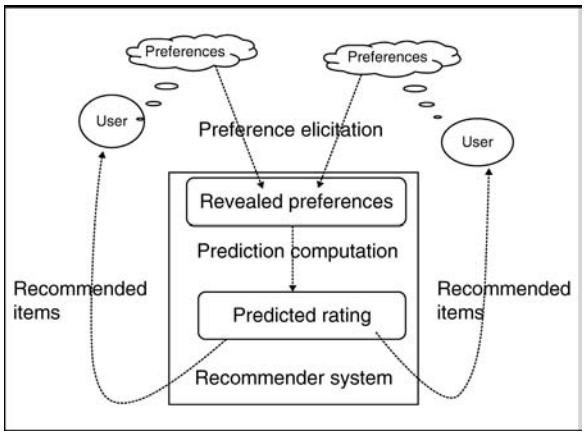

FIGURE 31.1 Schematic representation of a collaborative recommender system.

songs by "similar" artists recommended, where similarity is based on the collective preferences of people who listen to that artist.

To make recommendations like those at Last.fm, a collaborative recommender system must perform the following tasks:

1. Elicit Preferences: Learn about the users' preferences and store them in user profiles.
2. Compute Predictions: Predict how well a user would like an unfamiliar item based on the data from the user profiles.
3. Make Recommendations: Use the predictions to make recommendations.

Figure 31.1 provides a schematic representation of a recommender system, showing how these tasks relate to one another. Different recommender systems use different approaches to perform each of these tasks. We now discuss each of the tasks in turn.

Elicit Preferences

Recommender system can learn about the users' preferences by explicitly asking them to rate certain items; or by using implicit measures such as purchase history, search history, or time spent browsing an article. Most recommender systems that use the explicit method ask users to rate items in the database that they have experienced in the past. The systems use these ratings to form and periodically update a model of the user's preferences. Some recommender systems employ an alternative approach where they ask all users to rate items from the same "gauge" data set and model their preferences based on these ratings.[1]

Recommender systems can vary in the amount of detail that is captured with the ratings and the scale used to capture them (i.e., the dimensionality and granularity of ratings). A recommender system could obtain detailed ratings along multiple dimensions (e.g., quality and timeliness) or it could ask for a one-dimensional rating (e.g., overall satisfaction). Ratings could be on an all positive scale, all negative scale, or a positive and negative scale. The ratings can also vary on granularity. They may use a 1 to 5 scale like Amazon.com or a "thumbs up" and "thumbs down" scale like Digg. Each of these design choices has different implications for the accuracy of recommendations and also a user's privacy and the ease of entering ratings. They will also influence which prediction algorithms are possible to use.

© 2012 Taylor & Francis Group, LLC

Compute Predictions

Several alternative algorithms are used to make predictions and thus recommendations. Improving the accuracy of predictions is an area of active research, which has received additional impetus through Netflix's announcement in late 2006 of a $1 million prize for the first team that improves the predictions of Netflix's recommender algorithm by 10%.

Recommender algorithms can be classified into two types: (1) memory-based and (2) model-based.[2] Memory-based algorithms use data from all users to make predictions directly, while model-based algorithms use data from all users to formulate a single model of user preferences, and then use this model to make predictions. Recommender algorithms can also be classified based on whether they use correlations between users or between items.

In the user-user approach, correlation between two users is computed based on the scores of items that are rated (or used if the recommender system uses implicit measures) by both the users. The recommender system computes correlations between all such user pairs, which can be used in a variety of ways. One of the popular approaches is to use these correlations as weights when making predictions by taking a weighted average over the opinions of other users who have rated an item. Another approach is to use the user–user correlations to divide the user population into clusters of users, where users in the same cluster are considered to have similar preferences. Predictions for a user's hitherto unrated item are made by averaging the opinions of the other users in her cluster. Other approaches use statistical techniques such as Principal Component Analysis or Singular Value Decomposition and seek to identify latent factors in the data and make predictions based on those.

In systems that use the item–item approach (e.g., Amazon.com's "Users who bought this book also bought" feature), correlations between item pairs are computed instead of correlations between user pairs. Correlation between two items is computed using ratings of all the users who have rated both items. The system recommends items that are highly correlated with the items that are highly rated (or used) by the user.

When computing predictions, recommender algorithms usually perform some sort of normalization in order to account for systematic differences in the way people choose ratings. For example, if a person's average rating is 4.5 out of 5 and they give something a 3, the 3 rating is pretty bad. On the other hand, if a person has an average rating of 2.5 out of 5, a 3 rating is pretty good.

Make Recommendations

The objective of a recommender system is to present each user with items she is most likely to enjoy. To this effect, a recommender system can use the predictions in a variety of ways. As illustrated in the Last.fm example, the user can choose to have recommended songs play based on similar artists, neighbors, or her entire history.

System designers need to make several important design decisions in this phase. Besides the obvious design decisions about the interface, the system designer needs to determine what is the maximum permissible error. The error could be an error of commission (incorrectly recommending an item) or an error of omission (not recommending an item that should be recommended). The margin of permissible error for either type depends on the benefit of a correct recommendation and the cost of an incorrect recommendation. Consider a hypothetical Web site where medical treatment in discussed and recommended. Here a good recommendation not made or an incorrectly made recommendation could affect the health of a user. For video sharing, Web sites like YouTube, a recommendation is unlikely to have such important ramifications for the user, but it is still an important decision for the service provider. The service provider has an incentive to provide more recommendations if it leads to increased usage (or sale). At the same time the reduced usability due to poor recommendations may result in the user opting out of the system altogether, so there is still a need for an appropriate threshold.

The system designer needs to consider these issues when deciding the metrics for evaluating the predictions and the magnitude of permissible error. A good recommender system should continually seek feedback from the user and evaluate the accuracy of its predictions. Root Mean Square Error

© 2012 Taylor & Francis Group, LLC

(RMSE) and Mean Absolute Error are some of the evaluation metrics commonly used in recommender systems. The Netflix prize requires a 10% RMSE improvement in the predictions of Netflix's algorithm.

Content-Based Systems

Content-based systems recommend items based on features of the items themselves. Unlike collaborative systems, there is no need for data from other individuals. For a content-based system to work, a representation of each item must be generated. This can be done automatically, as when all of the words in a book or article are indexed. Or, it can be done manually as when a human cataloger associates a particular genre (i.e., romance) or subject heading to a book. An example of a very simple content-based system would be a news aggregation Web site that displays "related" articles that are textually "similar" to the one that a user is currently reading.

Many content-based systems provide personalized recommendations. In addition to having a representation of each item, this requires that individuals have a user profile that includes data about a user's likes and dislikes. As with collaborative systems, data for a content-based user profile can be explicitly entered or implicitly captured based on behavior (e.g., purchasing patterns).

This user profile is then compared with the representations of potential items and those that match closely are recommended. A variety of different techniques (i.e., algorithms) are used to compare user profiles and item representations in order to predict which items a user will like. These differ from those used in collaborative systems because the comparison is not between different ratings; it is between a user's profile and the representations of items. Common techniques are the use of information retrieval and machine learning algorithms (Hinshelwood[3] for a more complete list with examples).

Pandora is an example of a content-based system that recommends music. In contrast to Last.fm, it does not rely on any other user recommendations. Instead, it is based on a representation of each song, called a Music Genome, which is created automatically by a special software tool. A song's Music Genome consists of hundreds of musical attributes that describe qualities of melody, harmony, rhythm, form, composition, and lyrics. When a user enters a favorite song, Pandora recommends other songs with a similar Music Genome. Over time, users are presented with new songs which they rate. A user profile is automatically created that keeps track of the user's likes and dislikes. Additional songs are recommended taking into consideration the entire user profile.

Challenges of Recommender Systems

Designers and managers of recommender systems face several challenges. Some of these are dependent on the type of recommender system.

Content-based systems face two primary challenges:

1. Creating representations of certain items can be costly and time intensive. Although full-text, digital documents lend themselves well to automatic indexing, other items such as physical objects, small textual items such as quotes, and movies are difficult to automatically index in a satisfactory way.
2. Even when representations of items are available, they may not represent the characteristics that are most important to the user's enjoyment of the item. Jokes are a good example of this. While one could index the words used in a joke, the user's enjoyment of the joke has far less to do with the words than it does with the humor. In other words, knowing that a joke is about a chicken doesn't help a user know if she would enjoy the joke. One potential strategy for overcome this problem would be to add meta-data to items. However, for items like jokes or poems the meta-data would be so subjective that it would not likely produce accurate predictions for any particular user.[3]

© 2012 Taylor & Francis Group, LLC

Since collaborative systems are not based on the representation of an item, they can work for items that are costly or difficult to accurately represent, such as physical objects or jokes. However, they have their own set of challenges:

1. Eliciting enough user ratings to generate accurate predictions for all items is a constant source of concern for the designers and managers of recommender systems. Users need to be provided with sufficient incentives to participate.
2. New users and new items both suffer from the "cold-start" problem. New users do not get good recommendations until they have rated a sufficient number of items; and new items rarely get recommended until a sufficient number of users have rated (or used) them.
3. There may be entities that have an interest in manipulating the recommender system in order to promote certain items. An example of such a manipulation scheme called "sybil attack" or "shilling attack" involves creating a number of spurious users and providing ratings such that certain items get recommended more often. Preventing such manipulations or limiting the damage they cause, is an important consideration for recommender system designers and an active area of research.

To overcome some of these challenges, recommender systems may use a combined approach (both content-based and collaborative) or provide tools that allow people to use preexisting data (e.g., upload iTunes playlists all at once).

Recommender System Research

Online recommender systems have been developed and studied since the mid-1990s. Two related systems were simultaneously developed in the mid-1990s and were instrumental in showing the value of collaborative recommender systems:

1. GroupLens—a net news collaborative recommender system created by Resnick et al.[4] Resnick continued to study recommender systems as an editor for a special issue of *Communications of the Association for Computing Machinery (ACM)* on the topic in 1997[5] and as a contributor to numerous articles on the subject. Reidl has also remained highly active in recommender system research with his colleagues at the GroupLens research lab at the University of Minnesota. They have performed a number of studies of MovieLens, a recommender system for movies. Their Web site[6] is a good starting point for potential researchers with its list of publications and downloadable datasets.
2. Ringo—a music collaborative recommender system developed at MIT by Shardanand and Maes.[7] It was later made into a commercial product called Firefly, which was eventually bought out by Microsoft. Music recommender systems are now common and among the most advanced and popular (e.g., Last.fm, Pandora).

Research on recommender systems has continued to grow and doesn't show any signs of slowing down. Several special issues of well-respected journals have focused on recommender systems including the *Communications of the ACM*,[5] *ACM Transactions on Information Systems*,[8] *ACM Transactions on Computer–Human Interaction*,[9] and *IEEE Intelligent Systems*.[10] Research can also be found in conferences like ACM Special Interest Group on Information Retrieval, ACM Computer-Human Interaction, and ACM Electronic Commerce. Numerous workshops have been held over the years, and in 2007, the first annual ACM Recommender Systems[11] conference was held. Current research focuses on nearly every aspect of recommender systems from the recommendation algorithms, to interfaces design, to security, and privacy issues.

© 2012 Taylor & Francis Group, LLC

Summary

Recommender Systems are a powerful tool for recommending new items to individuals either based on content (content-based systems) or other users' experiences (collaborative systems). It is a highly active area of research that epitomizes the current social computing trends. While there are challenges with recommender systems (e.g., needing sufficient numbers of people and items rated before it works), they have already become widely used by corporations such as Amazon, NetFlix, Pandora, TiVo, Google, and others.

EXPERT LOCATOR SYSTEMS

Turning to experts for help is nothing new. We are all familiar with the ability of experts to diagnose a complex problem, clarify an issue, identify hidden structure, point us to a hard-to-locate resource, and perform a task that requires significant skill. Although some expert knowledge can be made explicit in the form of books, videos, diagrams, and knowledge-base entries, other knowledge is implicit and difficult to codify. Thus, it is often preferable to gain access to the source of the knowledge, the expert, in order to obtain the full benefit of the expertise. Unfortunately, it is not always easy to identify experts, especially within large organizations or distributed communities. Recently, systems have been developed to help locate individuals with needed expertise. These expert locator systems go by many names including expertise finders, expertise location engines, expert locators, and enterprise expertise management systems.

An expert locator system is a collection of technologies and social practices designed to help an individual find someone with the appropriate skills, knowledge, or expertise to meet a particular need. Some are stand-alone systems, but most are integrated into a more comprehensive knowledge management solution. While a basic organizational chart or an informal friendship-based network may be considered an expert locator system in the broadest sense, the term typically refers to more advanced systems that use implicitly or explicitly provided data to identify experts.

Researchers and practitioners have developed and examined expert locator systems since the early 1990s. Most empirical studies have taken place within large corporate settings, although more recent work has looked at expertise location among peer groups and virtual help-based communities. The most active research communities currently examining expert locator systems are the Computer Supported Cooperative Work (CSCW) and Knowledge Management communities. As a result, research on the topic is often published in conference proceedings in these areas (e.g., ACM–CSCW, ACM Conference on Information and Knowledge Management, ACM International Conference on Knowledge Discovery and Data Mining, ACM Recommender Systems) and information systems journals such as CSCW and KES (Knowledge-Based and Intelligent Engineering Systems). However, a considerable amount of research is scattered throughout publications on topics such as artificial intelligence, algorithms, Web personalization, and information systems more generally.

The following sections describe different subtypes of expert locator systems. The area is new enough that the vocabulary around them has not yet solidified. We group the systems into the following categories: *expert databases*, *automatic expertise finders*, *expertise recommenders*, and *expert referral systems*.

Expert Databases

Early expert locator systems were usually called expertise databases, knowledge directories, yellow pages, or knowledge maps. These systems consist of a searchable database of individuals along with data about their prior experience, expertise, organizational role, and contact information. Typical systems include Microsoft SPUD, HP CONNEX, and the NASA expertseeker. These systems are

© 2012 Taylor & Francis Group, LLC

usually designed for identifying experts to help solve technical problems or to match employee competencies with company positions.

Inputting accurate and detailed enough data into these databases can be a significant challenge. Some organizations rely upon assessment interviews, skill inventories, and extensive surveys of employees, but such methods can be costly and labor intensive. In other cases, individual employees are expected to enter information about themselves. Although individuals are the most qualified to describe their own expertise, they often lack motivation to add content—an activity that has few immediate rewards. Furthermore, they may not recognize the potential value of some of their less obvious skills and fail to report them. No matter who contributes the data, expert database entries can suffer from being over-simplified, one-dimensional assessments of expertise that are not informative enough to help direct the fine-grained, context specific questions that lead people to seek out experts. Finally, some systems rely on taxonomies to describe and catalog people's knowledge and skills. While this may encourage consistency and point out areas that may not have been considered, developing, and implementing taxonomies requires considerable effort and are likely to be misapplied if individuals are entering their own data.

Another related challenge is maintaining content over time. People leave, new skills are developed, positions change. For those who rarely use expert locator systems, keeping their data current is not on their top list of priorities. As a result, some expert databases quickly become obsolete. Additionally, organizations may not initially recognize the full investment required to maintain these systems once they are created.

Automatic Expertise Finders

As more and more activity occurs in the digital environment, it has become possible to profile individuals' expertise based on their conversations (e.g., in discussion forums and e-mail exchanges) and the documents associated with them (e.g., publications). An *automatic expertise finder* is a type of expert locator system that takes advantage of the implicit data left behind in the form of digital traces and documents. Such systems typically build expertise profiles from the implicit data by using information retrieval techniques (e.g., indexing). A person's expertise is usually described as a term vector and is used later for matching expertise queries using standard IR techniques. This allows people to search for a relevant expert in much the same way that they might search for a relevant document.

Well-known systems in this category include Who-Knows,[12] ContactFinder,[13] and MITRE MII Expert Finder.[14] Who-Knows identifies experts across an organization by using Latent Semantic Indexing techniques on the project documents people produce. ContactFinder identifies experts based on their participation patterns and message content. Expert Finder identifies experts based on documents people produce, as well as some experience-related information including basic employment information (e.g., positions held) and projects in which they participated.

These systems solve many of the challenges of expert databases since there is no need to manually contribute and maintain expertise information, and the automatically generated profiles are considerably more developed than the simple keyword-based profiles. However, these systems also have limitations. First, some individuals have expertise that is not yet represented in their digital traces. This is particularly true of new employees, as well as individuals who rely primarily on telephone and face-to-face meetings. Second, from the technical perspective, we still need to improve ways of selecting and integrating different sources and types of data to better reflect people's expertise. We also need to improve the ways of matching information seekers' fine-grained information needs with the large and amorphous expertise profiles. These are active areas of research, and we can expect improvements in the techniques that are used.

Finally, and perhaps most importantly, these systems largely do not consider the social perspectives of expertise sharing. For instance, their results are usually ranked purely based on the computed information similarity between the query and profiles. However, there are many other

© 2012 Taylor & Francis Group, LLC

criteria that people use to select experts in real life and many other social factors that contribute to individual's willingness to share information and have meaningful interactions. The following section discusses systems that were designed with these social considerations in mind.

Expertise Recommenders

Rooted in the field of CSCW, Ackerman and other researchers developed a series of systems that address both social and technical issues related to expertise location and sharing. In contrast to systems that only identify experts based on content overlap, these systems attempt to create a social and technical environment that encourages information sharing and recognizes the importance of social context.

Answer Garden (AG) is a system designed to help in situations like technical support, where there is a continuing stream of questions, many of which occur repeatedly, but some of which have never been seen before.[15] It has a branching network of diagnostic questions that helps users find answers. If there is no available answer, it automatically routes the question to the appropriate expert who can answer the user and record the answer into the branching network for future users. The design of AG addresses two important social issues in expertise finding. First, askers are anonymous to the experts. This decreases the asker's social costs related to status implications and the need for reciprocity, although it also loses some of the potentially helpful contextual information. Second, by continually adding questions and answers into the corpus, it decreases the expert's workload in answering the same questions repeatedly and grows the organizational memory incrementally.

Field studies of AG showed mixed results. Questioners appreciated the anonymity, but many of the answers they received were not at the appropriate level (e.g., an answer was too technical and lengthy). This finding suggests that expertise locator systems should route organizational members to individuals with the right level of expertise, not just to experts with the highest level of expertise. A future field study of an AG-like system highlighted some of the limitations of the system including frustration due to incomplete data and continually changing classification schemes.[16] The study also found that the AG approach is subject to the impact of the given division of labor and organizational micro-politics.

A new version of AG, AG2 was developed to overcome some of the original limitations.[17] Unlike AG, where the expert location occurred manually, an expertise location engine was developed for AG2. Various computer-mediated communication mechanisms are also added. One important social innovation was the fact that the AG2 expert locator algorithm prefers to "stay local" when selecting expertise to allow contextualization, a concept that was found useful in later systems as well. If a local expert is unavailable, the system supports an escalation process whereby the query is sent on to others until an answer is provided. Thus, the system helps gracefully overcome failures with initial expert recommendations. Another interesting change to AG2 is that the system tends to blur the dichotomy between experts and seekers, recognizing that individuals may be novices in some areas and experts in other areas.

Expertise Recommender (ER) is another system developed by McDonald and Ackerman in order to address issues identified from a field study of AG2.[18] The major contribution of this system is that it can select experts based on a range of social factors such as organizational closeness and workload, not just level of expertise. As more data about our actions and relationships become available online, many new possibilities for identifying an *appropriate* expert become viable.

It is important to recognize that the systems discussed in this section are research prototype systems that are not as widely used as those previously discussed. Although a framework for including additional factors into the expert identification algorithms has been developed, few modules have been implemented. Future research examining the social factors that should be considered (e.g., privacy considerations and motivational issues) when recommending experts seems promising. It short, this research shows that finding an expert is not enough. One must also understand the other social factors related to their willingness to participate and have enough contextual knowledge to help.

© 2012 Taylor & Francis Group, LLC

Expert Referral Systems

Another approach to identifying experts is to use a referral process, where an individual has his colleagues and friends introduce, or refer, an expert. This referral method has been used throughout time. However, as more information about our social relationships is made available in digital form, systems have been developed to augment our ability to get high quality referrals from our peers. We call these *expert referral systems.*

ReferralWeb was the first well-known system that utilized social network information to help individuals find and be introduced to experts on a particular topic.[19] In ReferralWeb, people's expertise are indexed based on individuals' publications. Social network information is extracted from the coauthorships or co-appearances in their Web pages. Experts are identified via traditional information retrieval techniques (as described in the *automatic expertise finders* section). Once identified, the information seeker is presented with visualizations of the network structure, and a list of referral chains that can be taken to get from your known peers to the desired expert. For example, it might show that my friend John knows Lucy who knows the expert Jack; likewise it would show other paths to Jack through different friends who know him.

Although not designed specifically for the purpose of finding experts, Yenta[20] helps individuals find others with similar interests—individuals who may be in the best position to provide expert advice. Yenta acts like a personal agent. It creates people's personal interest profiles by mining documents in their local machines. The profile is stored locally and uses inter-agent communication to find people who have information similar to the query, all the while protecting the actual content from being shared with others. Yenta also clusters people based on their shared interests to built social coalitions and provides tools to communicate with others in the same cluster. Thus, Yenta can be thought of as a recommender system as described earlier in this article. Other related systems include MARS[21] and SWIM.[22] Recently, with the advancement of social network theory research, there are increasing number of peer-to-peer applications designed to share knowledge and resources (e.g., files and contacts) through social networks, as well as commercial social network systems (e.g., spoke and visiblepath) that are designed to help people share contact information. As social networking sites such as Facebook and LinkedIn become ubiquitous, expert referral systems will be a natural fit.

These expert referral systems have several advantages and disadvantages. They support the age-old practice of finding information through social contacts that is familiar to all of us and socially acceptable. They provide added motivation for individuals to help those who seek them out; after all, an expert is more likely to provide help if they have been introduced by a mutual friend or perhaps even an automatic agent that has identified some hidden similarity. These systems are relatively easy to implement using basic peer-to-peer and information retrieval techniques. They also can provide contextual information about individuals based on their social network relationships, helping expert seekers make more educated decisions about whether or not it is worth contacting a particular expert. Unfortunately, systems like ReferralWeb increase the amount of time required to contact an expert; not only the information seeker's time but also the friends that are part of the referral chain. They may also decrease the pool of experts who are reachable in practice, since referral chains that are too long (or nonexistent) discourage contact with the expert. There is a need for more empirical studies of these systems to help reveal additional advantages and disadvantages.

Summary

In this section we have outlined a variety of different expert locator systems including expert databases, automatic expertise finders, ERs, and expert referral systems. Additional systems are described in Ackerman et al.[23] Because research on expert locator systems is still in its infancy, the specific terminology and categorization we provide is tentative and likely to change as new techniques are developed and integrated with other knowledge management and social software

© 2012 Taylor & Francis Group, LLC

programs. However, many of the key principles, trends, and design considerations discussed above are enduring. In this summary, we address two of the most prominent.

One common trend is the use of implicit data rather than explicitly entered data in expert locator systems. This trend is likely to continue as more and more of our activities are recorded in a digital environment and can potentially be used to identify our expertise domains, our social network ties, and other factors of interest (e.g., availability). For instance, Zhang et al.[24] explored ways of using people's asking-answering histories in online forums to infer expertise levels. Further developing tools to integrate these various data sources will be vital to improving expert locator systems. These issues are also important for designing recommender systems more generally.

Another common theme is the need to consider the social implications of expert locator systems. The *expertise recommenders* and the *expert referral systems* emphasize that locating the most knowledgeable individual on a topic is not enough. After all, the most knowledgeable individual may not be able to present information at the right level for the information seeker or may be too far removed from the local context to be of benefit. Furthermore, the most knowledgeable individuals may not have sufficient incentives to participate, especially if they are bombarded with questions from strangers. Systems that take into consideration these social factors and align the incentives of the various parties are far more likely to succeed in the long run. In this sense, expert locator systems may be better labeled expertise sharing systems.

CONCLUSION

In this entry we have discussed two types of systems that facilitate learning from other people in a highly personal way. Recommender systems provide personalized recommendations for individuals about items that they have not yet experienced for themselves. Expert locators help identify people who are knowledgeable on a topic, so they can personally engage with those who seek their expertise. These systems provide nice alternatives and complements to traditional information retrieval techniques. When finding a relevant document is not enough, expert locators provide access to individuals with expertise and recommender systems provide additional pointers to resources that may not have been considered in the original search query. Research on these topics is growing rapidly and the methods for performing them are improving as a result. In addition, as more and more activity is implicitly captured online, it is increasingly possible to improve these tools and apply them in new domains.

REFERENCES

1. Goldberg, K.; Roeder, T.; Gupta, D.; Perkins, C. Eigentaste: A constant time collaborative filtering algorithm. Inform. Ret. **2001**, *4* (2), 133–151.
2. Breese, J.; Heckerman, D.; Kadie, C. Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence*, Uncertainty in Artificial Intelligence, Madison, WI, July 24–26, 1998; Cooper, G., Moral, S., Eds.; Morgan Kaufman: San Francisco, 1998; 43–52.
3. Pazzani, M.J.; Billsus, D. Content-based Recommendation Systems. In *The Adaptive Web*; Brusilovsky, P., Kobsa, A., Nejdl, W., Eds.; Springer: Berlin/Heidelberg, Germany, 2007; 325–341. (http://www.springerlink.com/content/qq35wt68l6774261/for all info).
4. Resnick, P.; Iacovou, N.; Suchak, M.; Bergstrom, P.; Riedl, J. GroupLens: An open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work*, Computer Supported Cooperative Work, Chapel Hill, NC, October 22–26, 1994; ACM Press: New York, 1994; 175–186.
5. Crawford, D., Ed. Commun. ACM **1997**, *40*, (3).
6. http://www.grouplens.org/ (accessed February 2008).
7. Shardanand, U.; Maes, P. Social information filtering: Algorithms for automating "word of mouth". In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Conference on Human Factors in Computing Systems, Denver, CO, May 7–11, 1995; Association for Computing Machinery/Addison-Wesley: New York, 1995; 210–217.

© 2012 Taylor & Francis Group, LLC

8. Konstan, J.A., Ed.; Introduction to recommender systems: Algorithms and Evaluation. ACM Trans. Inform. Syst. (TOIS) **2004**, *22* (1), 1–4.
9. Riedl, J. Dourish, P. Introduction to the special section on recommender systems. ACM Trans. Comput.-Hum. Interact. (TOCHI) **2005**, *12* (3), 371–373.
10. Felfernig, A.; Friedrich, G.; Schmidt-Thieme, L. Guest editors'. Introduction: Recommender systems, IEEE Intell. Syst. **2007**, *22* (3), 18–21, doi: 10.1109/MIS.2007.52.
11. *Proceedings of the 2007 ACM Conference on Recommender Systems*, RecSys'07, Minneapolis, MN, October 19–20, 2007; ACM Press: New York, 2007.
12. Streeter, L.; Lochbaum, K. Who knows: A system based on automatic representation of semantic structure. In *Proceedings of the Conference on Computer-Assisted Information Retrieval*, RIAO'88 Program Conference, Cambridge, MA, March 21–24, 1988; CID: Paris, 1988; 380–388.
13. Krulwich, B.; Burkey, C. Contactfinder agent: Answering bulletin board questions with referrals. In *Proceedings of the 13th National Conference on Artificial Intelligence*, AAAI National Conference, Portland, OR, August 4–8, 1996; AAAI Press: Menlo Park, CA, 1996; 10–15.
14. Maybury, M.; D'Amore, R.; House, D. Automated discovery and mapping of expertise. In *Sharing Expertise: Beyond Knowledge Management*; Ackerman, M.S., Pipek, V., Wulf, V., Eds.; MIT Press: Cambridge, MA, 2003; 359–382.
15. Ackerman, M.S. Answer garden: A tool for growing organizational memory. Wirtschaftsinformatik **1995**, *37* (3), 320–321.
16. Pipek, V.; Wulf, V. Pruning the answer garden: Knowledge sharing in maintenance engineering. In *ECSCW 2003: Proceedings of the Eighth European Conference on Computer Supported Cooperative Work*, Computer Supported Cooperative Work, Helsinki, Finland, September 14–18, 2003; Kuutti, K., Karsten, E.H., Fitzpatrick, G., Dourish, P., Schmidt, K., Eds.; Kluwer Academic: Dordrecht, the Netherlands, 2003; 1–20.
17. Ackerman, M.S.; McDonald, D.W. Answer garden 2: Merging organizational memory with collaborative help. In *Proceedings of the 1996 ACM Conference on Computer Supported Cooperative Work*, Computer Supported Cooperative Work, Boston, November 16–20, 1996; ACM Press: New York, 1996; 97–105.
18. McDonald, D.W.; Ackerman, M.S. Expertise recom-mender: A flexible recommendation system and architecture. In *Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work*, Computer Supported Cooperative Work, Philadelphia, PA, December 2–6, 2000; ACM Press: New York, 2000; 231–240.
19. Kautz, H.; Selman, B.; Shah, M. Referral web: Combining social networks and collaborative filtering. Commun. ACM **1997**, *40* (3), 63–65.
20. Foner, L.N. Yenta: A multi-agent, referral-based matchmaking system. In *Proceedings of the 1st International Conference on Autonomous Agents*, International Conference on Autonomous Agents, Marina del Rey, CA, February 5–8, 1997; ACM Press: New York, 1997; 301–307.
21. Yu, B.; Singh, M.P. Searching social networks. In *Proceedings of the 2nd International Joint Conference on Autonomous Agents and Multiagent Systems*, International Conference on Autonomous Agents, Melbourne, Australia, July 14–18, 2003; ACM Press: New York, 2003; 65–72.
22. Zhang, J.; Van Alstyne, M. SWIM: Fostering social network based information search. In *CHI '04 Extended Abstracts on Human Factors in Computing Systems*, Conference on Human Factors in Computing Systems, Vienna, Austria, April 24–29, 2004; ACM Press: New York, 2004; 1568.
23. Ackerman, M.; Pipek, V.; Wulf, V., *Eds. Sharing Expertise: Beyond Knowledge Management*; MIT Press: Cambridge, MA, 2002.
24. Zhang, J.; Ackerman, M.S.; Adamic, L. Expertise networks in online communities: Structure and algorithms. In *Proceedings of the 16th International Conference on World Wide Web*, International World Wide Web Conference, Banff, Canada, May 8–12, 2007; ACM Press: New York, 2007; 221–230.

BIBLIOGRAPHY

1. Ackerman, M.S.; Halverson, C.A. Sharing expertise: The next step for knowledge management. In Social Capital and Information Technology; Huysman, M., Wulf, V., Eds.; MIT Press: Cambridge, MA, 2004; 273–300.
2. Ackerman, M.; Pipek, V.; Wulf, V., Eds. *Sharing Expertise: Beyond Knowledge Management*; MIT Press: Cambridge, MA, 2002.

© 2012 Taylor & Francis Group, LLC

3. Adomavicius, G.; Tuzhilin, A. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE T. Knowl. Data. En., **2005**, *17* (6), 734–749.
4. http://www.grouplens.org/ (accessed February 2008).
5. *Proceedings of the 2007 ACM Conference on Recommender Systems*, RecSys'07, Minneapolis, MN, October 19–20, 2007; ACM Press: New York, 2007.
6. Resnick, P.; Varian, H.R. Recommender systems. Comm. ACM **1997**, *40* (3), 56–58.
7. Riedl, J.; Konstan, J. *Word of Mouse: The Marketing Power of Collaborative Filtering*; Warner Books: New York, 2002.
8. Terveen, L.; Hill, W. Beyond recommender systems: Helping people help each other. In *HCI in the New Millennium*; Carroll, J., Ed.; Addison Wesley: Boston, 2001.

© 2012 Taylor & Francis Group, LLC

This page intentionally left blank

32 Knowledge Management Systems

Dick Stenmark

CONTENTS

INTRODUCTION

Knowledge management systems (KMS) refers to a class of information systems (IS) that is used to enhance knowledge and information transfer within an organization and to manage organizational knowledge.[1] Although this statement seems straightforward, it is rather difficult to define what a KMS is, since it is still unclear exactly what should be included in the concept of knowledge management (KM). What is the difference between knowledge and information and can computer applications really deal with knowledge? Some have argued that computers can only process data, and certainly not knowledge.[2] What, then, is a KMS and what role does it play in KM work? Before we can talk about *systems* for KM, we need a shared view of KM itself.

Is KM an emerging new discipline of its own or is it a topic that runs across several existing scholarly discourses? Throughout the years, there have been advocates of both positions. There are also commentators arguing that organizations have always been practicing KM-related activities so there is essentially nothing new to KM. This is no uncommon phenomenon but something that happens whenever new terminology is introduced. In the 1980s, Cronin asked whether information management had something new to offer or if it was just a new label for librarianship.[3] Twenty or so years later, people ask the same thing about KM and KMS.

Whilst several voices claim that there is no consensus regarding what exactly knowledge management is or how it differs from information management, Davenport and Grover in their editorial comment on the 2001 Special KM issue of Journal of Management Information Systems (JMIS)

© 2012 Taylor & Francis Group, LLC

concluded that KM as a *practical* phenomenon was here to stay but that *formal research* on the topic was lacking.[4] A couple of years have passed since Davenport and Grover made this observation and today it is fair to say that KM has established itself as a research topic, as reported elsewhere in this encyclopedia. Davenport and Grover continued by observing that IT support for KM, i.e., KMS, was seen as a useful but far from required resource by practitioners and scholars alike.

In the following text, we shall first look briefly at the history of KM and KMS before turning to the various theories underpinning this field. Thereafter, some of the most well-cited frameworks for KMS found in the IS literature are introduced to help the reader see what *types* of KMS are available. The ontological aspects of KMS are touched upon before finally discussing the challenges KMS are facing today.

ROOTS OF KM

It can be debated when and where KM started, since it depends on what discipline you examine. A large number of fields have clearly influenced the emerging KM discourse—e.g., sociology, human resource management, organization science, and IS research to name but a few. Many commentators would probably hold organizational learning as the one discipline that has had the perhaps most profound effect on the KM field.[5,6] When it comes to KMS, though, the IS discipline has taken a leading role, since the development, implementation, and use of systems to informate and automate are central to the IS field.[7]

Tiwana[8] places the roots of KM in the 1950s management literature, whereas Maier[6] traces the first instances of KM back to the studies of societal application of knowledge in the late 1960s and early 1970s. However, it was not until the late 1980s, through the writings of, e.g., Sveiby and Lloyd[9] and Wiig,[10] that the phenomenon started to receive more widespread attention. Nonaka and Takeuchis's book "The Knowledge Creating Company"[11] is also an early landmark in organizational KM literature and one of the most cited sources with almost 10,000 references in Google Scholar.

What propelled the development of KM as a new research discipline was the growing emphasis on knowledge work and knowledge workers as the primary source of productivity as opposed to assets such as land or capital.[11] This view paved way for the knowledge-based perspective of the firm[12] that suggests that the tangible resources of an organization generate value dependent on how they are combined and applied, which in turn depends on the organization's knowledge. This knowledge is deeply permeated in culture, procedures, routines, systems, and minds of the individual employees.

In the introduction of their 2001 paper, Alavi and Leidner note that it is not the existence of this knowledge per se that matters, but the ability to apply it and put it to use. To that end, advanced information technologies can be instrumental in effectuating the knowledge-based view of the firm by systemizing, enhancing, and expediting large-scale knowledge management.[13]

CRITIQUE OF KM

Knowledge management as a research discipline has also received critique. Some have argued that it is no more than yet another exemplar in a long list of management fads that have come and gone over the years.[2,14] In his critical analysis of KM, Wilson concludes:

> [Knowledge management] is, in large part, a management fad, promulgated mainly by certain consultancy companies, and the probability is that it will fade away like previous fads.[14]

Much of this skepticism stems from the fact that many consultancy firms and software vendors simply seemed to have renamed their old services and products, replacing the term "information" with the term "knowledge." Therefore, says Wilson:

> [T]he confusion of 'knowledge' as a synonym for 'information' is one of the most common effects of the 'knowledge management' fad.[14]

© 2012 Taylor & Francis Group, LLC

In addition to the fad debate, the KM discourse was also criticized for being "technology-driven." Comparing and contrasting the KM literature to that of organizational learning (OL), Swan et al. found that although the two disciplines are concerned with the improvement of organizational performance through knowledge development, i.e., human issues, only the OL literature focused on humans whereas the KM literature was predominantly occupied with tools and techniques.[5] The emphasis on information technology in the KM literature resulted in people being marginalized to either "inputs to KM [. . .] or as constraints on its effectiveness [. . .]" (p. 673).[5] Swan and colleagues argue that much of the richness of human relations is lost when KM is reduced to merely technology.

This distinction between technology-oriented and human-oriented approaches has a long tradition in organization science and goes back to at least the early 1980s. However, a more holistic understanding of KM that encompasses both these stances has developed, and much of the turf wars from the late 1990s have now abated.

THEORETICAL FOUNDATION

Much of the epistemology used in KM literature has been influenced by the separation of knowledge in a tacit and an explicit component. The notion of tacit knowing is attributed to philosopher Michael Polanyi but was introduced to the KM discourse by Nonaka and Takeuchi.[11] Interestingly, the commonly used tacit–explicit distinction is not directly derived from Polanyi's work. Most commentators see explicit knowledge as knowledge that has been captured and codified into manuals, procedures, and rules, and is easy to disseminate. Tacit knowledge, on the other hand, is then knowledge that cannot be easily articulated and thus only exists in people's hands and minds, and manifests itself through their actions. In contrast, Polanyi does not make such a distinction. Instead, he envisions tacit knowing as the backdrop against which all understanding is distinguished.

While Polanyi speaks of tacit *knowing*, i.e., the verb, as a backdrop against which all actions are understood, Nonaka and Takeuchi use the term tacit *knowledge*, i.e., the noun, to denote particular type of knowledge that is difficult to express. This view has been criticized but due to the strong influence of Nonaka and Takeuchi's writings on the knowledge management discourse, this interpretation has been widely adopted. Amongst the critics are Tsoukas, who argues that tacit knowledge is *not* explicit knowledge internalized. Instead, tacit knowledge is inseparable from explicit knowledge since "[t]acit knowledge is the necessary component of all knowledge" (p. 14).[15] According to Tsoukas the two are so inseparately related that to even try to separate the two would be to "miss the point." There had perhaps been less confusion had Nonaka instead used the term "implicit knowledge."

Tsoukas recognizes that the dichotomy between tacit and explicit knowledge and the taxonomies derived from this duality have advanced our understanding of organizational knowledge by showing its multifaceted nature. However, such typologies also limit our understanding by the inherent formalism that accompanies them. "The conceptual categories along which the phenomena are classified must be assumed to be discrete, separate, and stable. The problem is that they hardly ever are" (p. 14).[15]

The tacit–explicit dichotomy has also taken other expressions. Choo suggests a differentiation between tacit, explicit, and cultural knowledge,[16] and Spender suggests, in addition to tacit and explicit knowledge, individual and collective knowing.[12] Blackler speaks of embodied, embedded, embrained, encultured, and encoded knowledge.[17] Yet another derivative is the distinction between the community view and the commodity view. The community view sees knowledge as socially constructed and inseparable from the knower, whereas the commodity view holds knowledge as a universal truth, and as facts and collectable objects.[18] Though several other ways to classify knowledge exist and have been suggested, they all, more or less, build on the tacit–explicit dichotomy.

© 2012 Taylor & Francis Group, LLC

DEFINITION OF KM SYSTEMS

Whereas most people agree that data and information may exist outside humans, supporters of the community view of knowledge have argued that knowledge can never be separated from the knower and thus never stored digitally.[2,13] Computer support for knowledge management would thus be, in a sense, impossible. How can we then talk about KMS?

KMS is often employed as a catalyst or enabler of KM but such implementations should not be carried out without careful coordination with the required people-oriented activities needed. Alavi and Leidner note that while KM initiatives may not *require* tools, IT can certainly support KM in many ways, in particular in firms where the ratio of *knowledge workers* is high.[13] Schultze defines a knowledge worker as someone who interacts knowledgeable with information and sees information not only as something derived from knowledge but as something that changes knowledge.[19] There is thus a tight relationship between information and knowledge and it seems that any knowledge work needs to be supported by information technology.

As stated in the introduction, a KMS is an IS and IS and knowledge systems are thus not radically different; instead, there is a subtle yet important difference in the *attitude* towards and the *purpose* of the systems. Whereas an IS processes information without engaging the users, a KMS must be geared towards helping the users to understand and assign *meaning* to the information.[13] The value of any given piece of information resides in the relationship between the information and the user's knowledge. This means that design of KMS should be based on an understanding not only of the information per se, but also of the context where the user develops the information need, and the analysis of the usage of the same information once it has been obtained and interpreted by the user.[20]

Following Alavi and Leidner,[13] a KMS should thus be understood as a particular class of information systems developed specifically to support organisations in their attempt to create, codify, collect, store, integrate, share, and apply knowledge.

KMS THEORY

The theoretical foundation underpinning KMS vary considerably and are not easily detected but we can get a reasonably good picture by looking at Schultze and Leidner's classification of theoretical perspectives in KM-related IS research. Having reviewed six leading IS journals and thoroughly analyzed nearly 100 articles from 1990 to 2000, Schultze and Leidner showed that a vast majority or 70% belonged to the Normative Discourse, 25% could be labeled as Interpretative Discourse and only a handful of papers represented a Critical (or Dialogic) Discourse.[21]

According to Schultze and Leidner's analysis, the normative discourse, which is characterized by a strive towards consensus from an a priori understanding of what the research problems are, typically assumes progressive enlightenment and increasing rationalisation, management, and control. IS research representing the normative discourse are thus concerned with "codification, normalisation of experience and the search for law-like relationships" (pp. 216–217).[21] Much of the research focus on problem solving, and it creates "a problem space that can be decomposed in a logical, top-down fashion" (p. 221).[21] Although both the research topics and the way knowledge is operationalised show great diversity, a common metaphor used within the normative discourse is that of knowledge as an asset. Researchers in this category typically view knowledge as a key driver of organizational efficiency and performance. Amongst the theories underpinning normative research, Schultze and Leidner mention innovation diffusion theory, absorptive capacity theory, and management cognition theory (p. 222).[21]

The interpretative discourse, which also opts for consensus but from an emergent understanding of the organizational situation, emphasizes the social aspects of organizational life that has not been rationalized or systematized. IS research representing the interpretative discourse thus aims "to create a coherent, consensual, and unified representation of what the organisational reality is

© 2012 Taylor & Francis Group, LLC

'actually' like" (p. 217),[21] and is typically targeted on work situations and organizational practices. Knowledge is therefore studied indirectly via its role in organizational transformation and how it is supported by various types of KMS. In this discourse, knowledge, technology, and organisational practice are all seen as socially constructed and dynamic, and the theories upon which interpretative research rests include organisational learning, communities of practice, activity theory, and bricolage (pp. 224–225).[21]

It is evident that almost all KM-related IS research is consensus-oriented. There are, however, also those who apply a dissensus-oriented approach. Although Schultze and Leidner treat critical and dialogic as two separate discourses, I shall here use the critical discourse label to include both these perspectives, since both understand struggle, conflict, and tension as natural organizational states. Seen from this perspective, organizations are "sites of political struggle and fields of continuous conflict" (p. 217)[21] and the objective of the research is thus to show that there is no coherent reality but different forms of domination and distortions. KMS (and other IT tools) are thus not to be understood as neutral, according to this perspective, but should be seen as instruments to make invisible work visible or to actively change social conditions. Schultze and Leidner call for more research in the critical discourse since this perspective allows the highlighting of the social inequities underpinning the distinction between service and knowledge work and the examination of contradictions in managing knowledge.[21] The direct implications for KMS, however, are less obvious.

TYPES OF KMS AND THEIR APPLICATIONS

As we saw earlier, many vendors tried to repackage their applications under the KM label at the end of the last millennium and a list of different KMS can therefore be made arbitrarily long. Instead of presenting a list of software that not all would agree upon, and, in addition, soon would be dated, it is more useful to examine three of the most referenced classification schemes for KMS and let them define the *various types of applications* that are possible. The frameworks are Hansen et al.'s *Codification vs. Personalization* from *Harvard Business Review* in 1999,[22] Hahn and Subramani's *Knowledge Residence and Level of Structure* from ICIS 2000,[23] and finally Alavi and Leidner's scheme from MISQ in 2001.[13]

Codification vs. Personalization

An early framework for KM work (and hence for KMS to support that work) is found in Hansen et al.'s well-referred article from *Harvard Business Review.* Based on their studies of management consultancy firms, and implicitly building on Nonaka's dichotomy of explicit and tacit knowledge, Hansen et al. divide knowledge management efforts into two different strategies; codification and personalization.[22]

Companies where the KM strategy centers on codifying and storing knowledge into databases for easy dissemination and retrieval is said to follow a codification strategy. In such companies, computers have a central role in the strategy, as carriers of knowledge. Hansen et al. point to Ernst and Young as a company following a codification strategy. Knowledge is harvested and coded into documents or other "knowledge objects" as an informant called them (p. 108),[22] and these are thereafter stored in electronic repositories for later retrieval. Even though the codification process is laborious, Ernst and Young has dedicated staff members doing nothing else but codifying knowledge into documents—this approach allows for scaling up since the repositories are accessible for all employees worldwide and available around the clock. Once the object is put into the repository it can be used over and over again at a very low cost, provided it does not have to be modified. Companies using the codification strategy thus typically deal with problems where the same solution can be applied many times. The "economics of reuse" is what motivates the KM efforts in these companies, and the KMS used are typically document management systems and databases.[22]

© 2012 Taylor & Francis Group, LLC

In contrast, when knowledge is tied to the individual that developed it and thus cannot be stored in a database, it has to be shared through face-to-face interactions. The role of the computers is thus to facilitate communication between people. Companies with this approach are said to follow a personalization strategy, and Hansen et al. mention McKinsey as a company in this category. In their company, knowledge emerges out of dialogues between individuals and their IT focus is thus to enable interactions between employees. Part of McKinsey's KM strategy is to move people between offices to expand their networks. Even though face-to-face meetings are unequalled for sharing tacit knowledge, space and time distances may sometime prevent people from physical meetings. McKinsey thus engage e-mail and video conferencing equipment to communicate and allow employees tap into the expertise of their peers. Companies following a personalization strategy typically deal with unique problems that do not have clear solutions and where customized answers must be provided. In "experts economics" knowledge is tacit and cannot be systematized and made efficient. Instead, these companies charge much higher prices, and KMS used are expert finder systems and communications software.[22]

Hansen et al. stress that companies should not try to combine these two strategies but, based on their business strategy, select one as their main KM strategy and merely use the other as a complementary strategy.

Knowledge Residence and Level of Structure

Adding another dimension to the tacit–explicit dichotomy, Hahn and Subramani present a framework for KMS by looking on the one hand at where the knowledge is said to reside (i.e., in artifacts or in people) and on the other hand to the extent to which knowledge is said to impose or require an a priori structure. These axes form a two-by-two matrix hosting four different classes of KMS.[23]

- One is where the system manages knowledge artifacts that has an inherent structure or where the system imposes a structure on the artifacts. Formal document repositories and data warehouses belong to this class.
- A second class also requires an a priori structure but manages links to knowledgeable people. A competence database intended to let employees find colleagues with specific skills falls into this class.
- A third class does not impose any structure in particular and assumes that knowledge is codified into artifacts. Intranets where Web pages and documents are found through full-text indexing search engines belong to this class.
- Finally, a fourth class again requires no structure but provides means for employees to identify and communicate with local experts. Discussion forums and e-mail Listservs are systems in this class.

Hahn and Subramani identify three interesting challenges regarding KMS. First, balancing information overload and potential useful content involves the size and diversity of both the users and the content. When the knowledge resides in artifacts, more items means higher chances of being able to find what you need. Also when human resources are required, more users increase the possibilities of finding a knowledgeable coworker. The down side is that more information also means more unrelated or useless information, and more users typically generate more interactions and more e-mails, which blurs the picture. For the same reason is diversity useful, and no problem in highly structured environments, but when structures and shared vocabularies are lacking, diversity can easily get overwhelming.[23]

Second, balancing additional workload and accurate content addresses the issue of keeping KMS updated. Highly structured environments require considerable efforts to ensure the appropriateness of the structure, and this work often comes on top of the employees' ordinary work tasks. In more loosely structured systems motivation to share knowledge often comes in the form of higher social

© 2012 Taylor & Francis Group, LLC

status. The downside is that those who contribute and earn a reputation may end up being occupied answering people's questions and helping colleagues instead of doing their jobs.[23]

Third, balancing exploitation and exploration means being aware of the fact that reliance on existing solutions only may result in a competency trap.[24] A system that supports the exploitation of existing knowledge may provide short-term benefits but in the long run be detrimental to the organization. At the same time, a system preoccupied with generating new knowledge may prevent organizational members from learning and adding to the collective experience that exists in the organization.

Hahn and Subramani suggest the KMS should consider including agent technology, collaborative filtering methods, advanced visualization tools, in order to address the above challenges.[23]

Alavi and Leidner's Scheme

Without suggesting an explicit framework, Alavi and Leidner in their review of the literature discussing applications of IT to knowledge management efforts, identify three common approaches: Coding and sharing of best practice, Creation of knowledge directories, and Creation of knowledge networks (p. 114).[13]

Coding and sharing of best practice is one of the most common applications of KMS, according to Alavi and Leidner. The term "best practice" is typically used to refer to a superior or exemplary practice that leads to superior performance. By collecting and codifying stories that mediate such practice, organizations can build KMS that stores and disseminates these experiences within the organization.

Creation of knowledge directories forms a second common class of KMS. Knowledge directories are also known as expert finder systems and aim at mapping the internal expertise of the organization. Alavi and Leidner report that 74% of the respondents in Gazeau's survey believed that their organization's most useful knowledge was not available in explicit form. When knowledge cannot be codified into artifacts, creating knowledge directories allows organizational members to benefit from the knowledge by being able to find and subsequently talk to the knowledgeable coworker.

Creation of knowledge networks is the third commonly used approach to KMS. Applications to first identify and then bring together (virtually or face-to-face) people from the same community of practice or those who share an interest has proven useful in many organizations. Ford Motor Company found that by sharing knowledge in networks the development time for cars was reduced by 33%. Online forums belong to the technology used in this approach.[13]

We have seen that KMS can either be used to support a commodity view of KM, where the explication of knowledge is assumed not only to be possible but also necessary, or a community view of KM, where the implicit nature of knowing puts people in focus. The success of KMS (as with most IS) depends on the extent of use, which in turn depends on a number of factors. In their concluding discussion, Alavi and Leidner point to a set of research questions concerning the application of IT to KM. In sum, they ask what effect an increased breadth and depth of knowledge via KMS would have on organizational performance; how to ensure that knowledge in an KMS can be modified (if necessary) prior to being used, and how these modifications too can be captured; how anonymized knowledge in a KMS can be trusted; and what are the quality and usefulness factors of KMS.[13] The answers to many of these questions are still pending.

ONTOLOGICAL ASPECTS

It has often been argued that only individuals can think and act—not organizations. At the same time, as human beings we are social creatures and we tend to seek, and benefit from, each other's company. Inputs from colleagues and the surrounding context greatly affect our ability to create and use knowledge because the individual and the collective interact in fruitful ways. Focusing primarily on how new knowledge emerges, Nonaka and Takeuchi stress the fact that knowledge creation

© 2012 Taylor & Francis Group, LLC

initiates from the individual but is a process that moves through expanding communities of interaction, crossing group, division, and, finally, organizational boundaries.[11] Other scholars have made similar comments about other KM processes.

Still, IT support for KM has traditionally focused on organizational-wide systems, possibly due to the acknowledged fact that the usefulness of a KMS grows exponentially with the size of the organization. Much of the IS research has thus had a macro-level focus, but also applications supporting organizational learning and organizational memory are common in the KMS repertoire. The challenge associated with organizational KMS is that individuals often have to provide input without getting much back in return. This problem, often referred to as the maintenance problem seriously threatens the quality and usefulness of these systems.

Another category of KMS are the groupware systems targeting smaller subsets of the organization, typically aiming for management. This category includes various types of Decision Support Systems (DSS). Many KMS in this category can also be related to the field of Computer-Supported Collaborative Work (CSCW). Typical applications here include Helpdesk applications and expert finder systems within specific subgroups. The maintenance problem continues to be a challenge also at this level.

When it comes to the individual, there has—until recently—not been equally much support. Some argue that this situation is about to change. One of the problems here is that not all of the applications used at an individual level are officially labeled KMS. For example, the information retrieval (IR) field has provided the knowledge worker with search engines and other tools to help locate information, but not all would agree that a search engine is a KMS. Another noticeable trend is the growth of social media. These applications exploit the individual–collective relationship and are able to provide the individual with added value through the actions of the collective relationship and vice versa. It will be interesting to follow this development to see whether social media will provide a means to avoid, if not solve, the maintenance problem.

THE KMS CHALLENGES

A number of KMS challenges can be identified in the KM literature. One issue is that of dispersion of work. It is argued that knowledge workers are increasingly dispersed—spatial as well as contractual.[25] Organizational members work outside the physical boundaries of the firm and/or change positions within the firm, often including geographical changes. This, it is argued, makes them less exposed to colleagues with similar functional skills.

There is also the contractual dispersion, i.e., the provisional nature of employment and the higher level of partial or temporary involvement in the firm that many knowledge workers experience. In addition, many are engaged in virtual teams that often reorganize and have high turnover rates. This dispersion of work requires KMS that allows for effective sharing of the latest knowledge.[25,26]

Another issue is the shorter product and process life cycles in today's organizations.[26] This compresses the time window for capturing the lessons learned and knowledge created in the process and leaves the knowledge workers with little time to document and save their experiences. At the same time knowledge becomes obsolete much quicker. KMS need to be able to deal effectively with these circumstances.

The above concerns can be seen as aspects of a larger and overarching challenge, i.e., how to keep KMS updated and current. While many of today's organizations expect KMS to become major catalysts for innovations in terms of the ways in which businesses can be organized and conducted, there is plenty of IS research that indicate that such systems often fail when implemented in everyday knowledge work. In response, a distinguishable issue in KMS research is how to support knowledge work with IS in a successful way. It has been found that although the systems work technically and should function well in theory, they remain unused by the organizational members.[27] Following this, the development of systems with the capacity to bridge the knowing-doing gap in organizations has been recognized as a significant area of KMS research.

© 2012 Taylor & Francis Group, LLC

However, the imbalance between the desire for accurate content and the workload required to achieve this still appears to be a critical problem, leading to systems of little use for organizations in their knowledge application processes. It has been suggested that the problem stems from the fact that the requirements for KMS are fundamentally different from those of other types of IT and are thus not covered by existing IS design theories.

Markus et al. have identified three primary differences.[28] First, knowledge work processes requires that expert knowledge is adapted and/or contextualized to specific local conditions. Decision support systems and executive IS do not provide system features that can handle expert knowledge or contextualize translation rules. Resulting from this, DSS and expert systems inhibit creative problem finding and solution generation. While expert systems manage general expert knowledge, they fail to support contextual knowledge and the flexibility needed for process emergence. Second, these types of systems are all specifically designed for a known type of user, e.g., managers. Being designed for a particular type of user community, these systems are not well adapted to emergent work processes characterized by shifting user types having varying knowledge requirements. Third, knowledge workers have access to many different types of systems but since these systems often are isolated and not integrated into work practice, knowledge workers tend to manage their systems rather than getting the job done.

To circumvent these problems, it has been suggested that KMS should be integrated with or build into already existing applications since key to a successful KMS is to facilitate usage.[29] As knowledge work requires creativity in order to produce idiosyncratic and esoteric knowledge, knowledge work practice is untidy compared to operational or administrative business processes. Hence, KMS must be able to go beyond written instructions and official task descriptions, thus appreciating exceptions not only as something inevitable but as a necessity. Consequently, KMS must not be isolated but should be integrated into work practice. For the purpose of avoiding situations where knowledge workers manage their systems rather than getting the job done, developers must recognize sociotechnical issues associated with disparity in work and benefit. In this way, KMS capable of attracting a critical mass of users can be developed. In addition, paying attention to unobtrusive accessibility and the adoption process may deepen developers' understanding of how support systems can be better integrated with both the day-to-day tasks of knowledge workers and their performance of the tasks.

CONCLUDING SUMMARY

In the 1990s there was a rather heated debate whether or not KM was a fad but this seems to have abated. Now, there is consensus that KM—at least as a pragmatic issue—is here to stay. With knowledge replacing economy of scale as business driver and with increasing portion of knowledge workers in today's organizations, knowledge management, and the need for IT support for it, is not likely to go away.

The strong focus on technology that we witnessed in early KM work has been compensated for and practitioners and researchers alike now have acknowledged that knowledge cannot exist outside the mind of a human being. Cultures that encourage and motivate individuals to share, combine, and reuse knowledge are recognized as equal, if not more, important as IT, even amongst technologists. IT is still likely to continue to play an important part, not as driver and single success factor but as catalyst, facilitator and enabler of social networks, virtual meeting places, and new discussion forums. One of the general lessons learned is that technology is important and useful but it should not be the driving force in KM work.

Several commentators have pointed to the fact that KMS in the late 1990s were discrete, standalone systems not aligned with the organizations' business processes. Such systems had to be explicitly attended to on top of ordinary tasks, thus adding to—not facilitating—the work to be carried out.[29] Newer KMS appear to be better integrated with existing business infrastructure and enterprise applications, thereby allowing employees to seamlessly apply organizational knowledge

© 2012 Taylor & Francis Group, LLC

in whatever work they are engaged. However, there is still a need for development and research in this area.

On the theoretical side, no core theory on knowledge management has yet been developed, and KM may still be understood as an "umbrella construct," i.e., a broad and somewhat unclear label that is used to contain a whole variety of loosely connected issues.[30] Without a clear theoretical focus, some commentators argue, the original concept risks being eroded until it has no value and collapses, as researchers explore divergent paths and build isolated islands of knowledge. Spender[31] has argued strongly that KM and KMS research need a core theory that distinguishes them from other fields but at the same time is narrow enough to allow laypeople to recognize and understand what is and what is not a KMS. Not much work is currently to be found along such lines.

In their editorial introduction to the 2003 special issue on KM and IT in *IT and People*, Gray and Meister argue that KMS researchers are facing a bigger problem than did researchers of earlier organizational phenomena, since knowledge is neither new nor physically present and there is thus nothing concrete to point to. An independent core theory of KM and KMS is therefore needed, they argue.[32]

However, several future scenarios are possible. If the development towards more knowledge work continues, we may end up in the scenario predicted by Davenport and Grover where "every industry will view itself as knowledge-intensive (p. 4).[4] If everything is KM, will the concept then still be meaningful, and if every application is a KMS, will the term be useful? At the other end of the spectrum lies a scenario where KM becomes so diversified and scattered that for this reason is pointless to talk about KM and IT support for it. Where we will end up remains to be seen.

REFERENCES

1. Voelpel, S.; Dous, M.; Davenport, T. Five steps to creating a global knowledge-sharing system: Siemens' ShareNet. Acad. Manage. Execut. **2005**, *19* (2), 9–23.
2. Galliers, R.; Newell, S. Back to the future: From knowledge management to data management. In *Proceedings of European Conference on Information Systems 2001*, Bled, Slovenia, June 27–29, 2001; 609–615.
3. Cronin, B. Introduction. *In Information Management: From Strategies to Action*; Cronin, B., Ed.; Aslib: London, 1985; vii–ix.
4. Davenport, T.H.; Grover, V. Editorial: Special issue on knowledge management. J. Manage. Inform. Syst. **2001**, *18* (1), 3–4.
5. Swan, J.; Scarbrough, H.; Preston, J. Knowledge management—The next fad to forget people. In *Proceedings of European Conference on Information Systems 1999*, Copenhagen, Denmark, June 23–25, 1999; 668–678.
6. Maier, R. *Knowledge Management Systems*, 2nd Ed.; Springer: Berlin, 2004.
7. Butler, T. From data to knowledge and back again: Understanding the limitations of KMS. Knowl. Process Manage. **2003**, *10* (3), 144–155.
8. Tiwana, A. *The Knowledge Management Toolkit: Practical Techniques for Building Knowledge Management Systems*; Pearson Education: Upper Saddle River, NJ, 1999.
9. Sveiby, K.E.; Lloyd, T. *Managing Know-How*; Bloomsbury: London, 1987.
10. Wiig, K.M. Management of knowledge: Perspectives of a new opportunity. In *User interfaces: Gateway or bottleneck?*; Bernold, T., Ed.; Gottlieb Duttweiler Institute: Zurich, 1988; 101–116.
11. Nonaka, I.; Takeuchi, H. *The Knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation*; Oxford University Press: Oxford, 1995.
12. Spender, J.-C. Making knowledge the basis of a dynamic theory of the firm. Strateg. Manage. J. Winter Special Issue, **1996**, *17*, 45–62.
13. Alavi, A.; Leidner, D. Review: Knowledge management and knowledge management systems: Conceptual foundations and research issues. MIS Q. **2001**, *25* (1), 107–136.
14. Wilson, T.D. The nonsense of knowledge management. Inform. Res. **2002**, *8* (1), paper no. 144. http://informationr.net/ir/8–1/paper144.html (accessed December. 2008).
15. Tsoukas, H. The firm as a distributed knowledge system: A constructionist approach. Strateg. Manage. J. Winter Special Issue **1996**, (17), 11–25.

© 2012 Taylor & Francis Group, LLC

16. Choo, C.W. *The Knowing Organization*; Oxford University Press: Oxford, 1998.
17. Blackler, F. Knowledge, knowledge work and organizations: an overview and interpretation. Organ. Stud. **1995**, *16* (6), 1021–1046.
18. Swan, J.; Scarbrough, H. Knowledge management: Concepts and controversies. J. Manage. Stud. **2001**, *38* (7), 913–921.
19. Schultze, U. A confessional account of an ethnography about knowledge work. Manage. Inform. Syst. Quart. **2000**, *24* (1), 3–41.
20. Stenmark, D. Information vs. knowledge: The role of intranets in knowledge management. In *Proceedings of HICSS-35*, Hawaii, January 7–10, 2002.
21. Schultze, U.; Leidner, D. Studying knowledge management in information systems research: Discourses and theoretical assumptions. Manage. Inform. Syst. Quart. **2002**, *26* (3), 213–242.
22. Hansen, M.; Nohria, N.; Tierney, T. What's your strategy for managing knowledge? Harvard Bus. Rev. March–April **1999**, *77* (2), 106–116.
23. Hahn, J.; Subramani, M. A framework of knowledge management systems: Issues and challenges for theory and practice. In *Proceedings of International Conference on Information Systems 2000*, Brisbane, Australia, December 10–13, 2000, 302–312.
24. Levitt, B.; March, J.G. Organizational learning. Annu. Rev. Sociol. **1988**, *14*, 319–340.
25. Corso, M.; Martini, A.; Pellegrini, L.; Massa, S.; Testa, S. Managing dispersed workers: The new challenge in knowledge management. Technovation **2006**, *26* (5–6), 583–594.
26. Donnellan, B.; Fitzgerald, B. Developing systems to support organisational learning in product development organisations. Electron. J. Knowl. Manage. **2003**, *1* (2), 33–46.
27. Schultze, U.; Boland, R.J. Knowledge management technology and the reproduction of knowledge work practices. J. Strateg. Inform. Syst. **2000**, *9*, 193–212.
28. Markus, L.M.; Majchrzak, A.; Gasser, L. A design theory for systems that support emergent knowledge processes. Manage. Inform. Syst. Quart. **2002**, *26*, 179–212.
29. Stenmark, D.; Lindgren, R. System support for knowledge work: Bridging the knowing-doing gap. Intl. J. Knowl. Manage. **2006**, *2* (2), 46–68.
30. Hirsch, P.; Levin, D. Umbrella advocates versus validity police: A life-cycle model. Organ. Sci. **1999**, *10*, 199–212.
31. Spender, J.-C. Exploring uncertainty and emotion in the knowledge-based theory of the firm. Inform. Technol. People **2003**, *16* (3), 266–288.
32. Gray, P.H.; Meister, D.B. Introduction: Fragmentation and integration in knowledge management research. Inform. Technol. People **2003**, *16* (3), 259–265.

BIBLIOGRAPHY

General KM

1. Davenport and Prusak, *Working Knowledge: How Organizations Manage What They Know*; Harvard Business School Press: Boston, 1997.

Knowledge Management Systems

2. Barnes, S., Ed.; *Knowledge Management Systems: Theory and Practice*; Thomson learning: London, 2002.
3. Malhotra, Y. Why knowledge management systems fail? Enablers and constraints of knowledge management in human enterprises. In *Handbook on Knowledge Management 1: Knowledge Matters*; Holsapple, Ed.; Springer-Verlag: Berlin, 2002; 577–599.
4. Rubenstein, A.H.; Geisler, E. *Installing and Managing Workable Knowledge Management Systems*; Greenwood Publishing Group Inc.: Westport, CT, 2003.
5. Ruggles, R.L., Ed.; *Knowledge Management Tools*; Butter-worth Heinemann: Boston, 1997.

© 2012 Taylor & Francis Group, LLC

This page intentionally left blank

33 Decision Support Systems

Marek J. Druzdzel and Roger R. Flynn

CONTENTS

INTRODUCTION

Making decisions concerning complex systems (e.g., the management of organizational operations, industrial processes, or investment portfolios; the command and control of military units; the control of nuclear power plants) often strains our cognitive capabilities. Even though individual interactions among a system's variables may be well understood, predicting how the system will react to an external manipulation such as a policy decision is often difficult. What will be, for example the effect of introducing the third shift on a factory floor? One might expect that this will increase the plant's output by roughly 50%. Factors such as additional wages, machine weardown, maintenance breaks, raw material usage, supply logistics, and future demand also need to be considered, however, because they will all affect the total financial outcome of this decision. Many variables are involved in complex and often subtle interdependencies, and predicting the total outcome may be daunting.

There is a substantial amount of empirical evidence that human intuitive judgment and decision making can be far from optimal, and it deteriorates even further with complexity and stress. In many situations, the quality of decisions is important; therefore, aiding the deficiencies of human judgment and decision making has been a major focus of science throughout history. Disciplines such as statistics, economics, and operations research developed various methods for making rational choices. More recently, these methods, often enhanced by various techniques originating from

© 2012 Taylor & Francis Group, LLC

information science, cognitive psychology, and artificial intelligence, have been implemented in the form of computer programs, either as stand-alone tools or as integrated computing environments for complex decision making. Such environments are often given the common name of *decision support systems* (DSSs). The concept of DSS is extremely broad, and its definitions vary, depending on the author's point of view. To avoid exclusion of any of the existing types of DSSs, we define them roughly as interactive computer-based systems that aid users in judgment and choice activities. Another name sometimes used as a synonym for DSS is *knowledge-based systems*, which refers to their attempt to formalize domain knowledge so that it is amenable to mechanized reasoning.

Decision support systems are gaining an increased popularity in various domains, including business, engineering, the military, and medicine. They are especially valuable in situations in which the amount of available information is prohibitive for the intuition of an unaided human decision maker, and in which precision and optimality are of importance. Decision support systems can aid human cognitive deficiencies by integrating various sources of information, providing intelligent access to relevant knowledge, and aiding the process of structuring decisions. They can also support choice among well-defined alternatives and build on formal approaches, such as the methods of engineering economics, operations research, statistics, and decision theory. They can also employ artificial intelligence methods to heuristically address problems that are intractable by formal techniques. Proper application of decision-making tools increases productivity, efficiency, and effectiveness, and gives many businesses a comparative advantage over their competitors, allowing them to make optimal choices for technological processes and their parameters, planning business operations, logistics, or investments.

Although it is difficult to overestimate the importance of various computer-based tools that are relevant to decision making (e.g., databases, planning software, spreadsheets), this entry focuses primarily on the core of a DSS, the part that directly supports modeling decision problems and identifies best alternatives. We briefly discuss the characteristics of decision problems and how decision making can be supported by computer programs. We then cover various components of DSSs and the role that they play in decision support. We also introduce an emergent class of *normative systems* (i.e., DSSs based on sound theoretical principles), and in particular, decision-analytic DSSs. Finally, we review issues related to user interfaces to DSSs and stress the importance of user interfaces to the ultimate quality of decisions aided by computer programs.

DECISIONS AND DECISION MODELING

Types of Decisions

A simple view of decision making is that it is a problem of choice among several alternatives. A somewhat more sophisticated view includes the process of constructing the alternatives (i.e., given a problem statement, developing a list of choice options). A complete picture includes a search for opportunities for decisions (i.e., discovering that there is a decision to be made). A manager of a company may face a choice in which the options are clear (e.g., the choice of a supplier from among all existing suppliers). She may also face a well-defined problem for which she designs creative decision options (e.g., how to market a new product so that the profits are maximized). Finally, she may work in a less reactive fashion, and view decision problems as opportunities that have to be discovered by studying the operations of her company and its surrounding environment (e.g., how can she make the production process more efficient). There is much anecdotal and some empirical evidence that structuring decision problems and identifying creative decision alternatives determine the ultimate quality of decisions. Decision support systems aim mainly at this broadest type of decision making, and in addition to supporting choice, they aid in modeling and analyzing systems (e.g., as complex organizations), identifying decision opportunities, and structuring decision problems.

© 2012 Taylor & Francis Group, LLC

Human Judgment and Decision Making

Theoretical studies on rational decision making, notably that in the context of probability theory and decision theory, have been accompanied by empirical research on whether human behavior complies with the theory. It has been rather convincingly demonstrated in numerous empirical studies that human judgment and decision making are based on intuitive strategies, as opposed to theoretically sound reasoning rules. These intuitive strategies, referred to as *judgmental heuristics* in the context of decision making, help us in reducing the cognitive load, but alas at the expense of optimal decision making. Effectively, our unaided judgment and choice exhibit systematic violations of probability axioms (referred to as *biases*). Formal discussion of the most important research results, along with experimental data, can be found in an anthology edited by Kahneman, Slovic, and Tversky.[1] Dawes[2] provided an accessible introduction to what is known about people's decision-making performance.

One might hope that people who have achieved expertise in a domain will not be subject to judgmental biases and will approach optimality in decision making. Although empirical evidence shows that experts indeed are more accurate than novices, within their area of expertise, it also shows that they also are liable to the same judgmental biases as novices, and demonstrate apparent errors and inconsistencies in their judgment. Professionals such as practicing physicians use essentially the same judgmental heuristics and are prone to the same biases, although the degree of departure from the normatively prescribed judgment seems to decrease with experience. In addition to laboratory evidence, there are several studies of expert performance in realistic settings, showing that it is inferior even to simple linear models (an informal review of the available evidence and pointers to literature can be found in the book by Dawes).[2] For example, predictions of future violent behavior of psychiatric patients made by a panel of psychiatrists who had access to patient records and interviewed the patients were found to be inferior to a simple model that included only the past incidence of violent behavior. Predictions of marriage counselors concerning marital happiness were shown to be inferior to a simple model that just subtracted the rate of fighting from the rate of sexual intercourse (again, the marriage counselors had access to all data, including interviews with the couples). Studies yielding similar results were conducted with bank loan officers, physicians, university admission committees, and so on.

Modeling Decisions

The superiority of even simple linear models over human intuitive judgment suggests that one way to improve the quality of decisions is to decompose a decision problem into simpler components that are well defined and well understood. Studying a complex system built out of such components can be subsequently aided by a formal, theoretically sound technique. The process of decomposing and formalizing a problem is often called modeling. Modeling amounts to finding an abstract representation of a real-world system that simplifies and assumes as much as possible about the system, and while retaining the system's essential relationships, omits unnecessary detail. Building a model of a decision problem, as opposed to reasoning about a problem in a holistic way, allows for applying scientific knowledge that can be transferred across problems and often across domains. It allows for analyzing, explaining, and arguing about a decision problem.

The desire to improve human decision making provided motivation for the development of various modeling tools in disciplines of economics, operations research, decision theory, decision analysis, and statistics. In each modeling tool, knowledge about a system is represented by means of algebraic, logical, or statistical variables. Interactions among these variables are expressed by equations or logical rules, possibly enhanced with an explicit representation of uncertainty. When the functional form of an interaction is unknown, it is sometimes described in purely probabilistic terms (e.g., by a conditional probability distribution). Once a model has been formulated, various mathematical methods can be used to analyze it. Decision making under certainty has been

© 2012 Taylor & Francis Group, LLC

addressed by economic and operations research methods, such as cash flow analysis, break-even analysis, scenario analysis, mathematical programming, inventory techniques, and various optimization algorithms for scheduling and logistics. Decision making under uncertainty enhances the above methods with statistical approaches, such as reliability analysis, simulation, and statistical decision making. Most of these methods have made it into college curricula and can be found in management textbooks. Due to space constraints, we do not discuss their details further.

Components of Decision Models

Although a model mathematically consists of variables and a specification of interactions among them, from the point of view of decision making, a model and its variables represent the following three components: 1) a measure of preferences over decision objectives; 2) available decision options; and 3) a measure of uncertainty over variables influencing the decision and the outcomes.

Preference is widely viewed as the most important concept in decision making. Outcomes of a decision process are not all equally attractive, and it is crucial for a decision maker to examine these outcomes in terms of their desirability. Preferences can be ordinal (e.g., more income is preferred to less income), but it is convenient and often necessary to represent them as numerical quantities, especially if the outcome of the decision process consists of multiple attributes that need to be compared on a common scale. Even when they consist of just a single attribute but the choice is made under uncertainty, expressing preferences numerically allows for trade-offs between desirability and risk.

The second component of decision problems is available decision options. Often these options can be enumerated (e.g., a list of possible suppliers), but sometimes they are continuous values of specified policy variables (e.g., the amount of raw material to be kept in stock). Listing the available decision options is an important element of model structuring.

The third element of decision models is uncertainty. Uncertainty is one of the most inherent and most prevalent properties of knowledge, originating from incompleteness of information, imprecision, and model approximations made for the sake of simplicity. It would not be an exaggeration to state that real-world decisions not involving uncertainty either do not exist or belong to a truly limited class. As Benjamin Franklin expressed it in 1789 in a letter to his friend M. Le Roy, "in this world nothing can said to be certain, except death and taxes" (*The Complete Works of Benjamin Franklin*, John Bigelow (Ed.), G.P. Putnam's Sons: New York and London, 1887; Vol. 10, 1700).

Decision making under uncertainty can be viewed as a deliberation—determining what action should be taken that will maximize the expected gain. Due to uncertainty, there is no guarantee that the result of the action will be the one intended, and the best one can hope for is to maximize the chance of a desirable outcome. The process rests on the assumption that a good decision is one that results from a good decision-making process that considers all important factors and is explicit about decision alternatives, preferences, and uncertainty.

It is important to distinguish between good decisions and good outcomes. By a stroke of good luck, a poor decision can lead to a very good outcome. Similarly, a very good decision can be followed by a bad outcome. Supporting decisions means supporting the decision-making process so that better decisions are made. Better decisions can be expected to lead to better outcomes.

DECISION SUPPORT SYSTEMS

Decision support systems are interactive, computer-based systems that aid users in judgment and choice activities. They provide data storage and retrieval, but enhance the traditional information access and retrieval functions with support for model building and model-based reasoning. They support framing, modeling, and problem solving.

Typical application areas of DSSs are management and planning in business, health care, the military, and any area in which management will encounter complex decision situations. Decision support

© 2012 Taylor & Francis Group, LLC

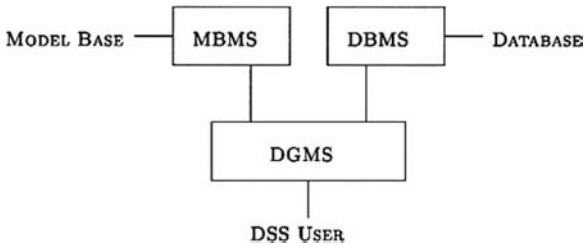

FIGURE 33.1 The architecture of a DSS. (From Sage, A.P. *Decision Support Systems Engineering*; John Wiley & Sons, Inc.: New York, 1991.[3])

systems are typically used for strategic and tactical decisions faced by upper-level management—decisions with a reasonably low frequency and high potential consequences—in which the time taken for thinking through and modeling the problem pays off generously in the long run.

There are three fundamental components of DSSs[3]:

- *Database management system* (*DBMS*). A DBMS serves as a data bank for the DSS. It stores large quantities of data that are relevant to the class of problems for which the DSS has been designed and provides logical data structures (as opposed to the physical data structures) with which the users interact. A DBMS separates the users from the physical aspects of the database structure and processing. It should also be capable of informing the user of the types of data that are available and how to gain access to them.
- *Model-base management system* (*MBMS*). The role of MBMS is analogous to that of a DBMS. Its primary function is providing independence between specific models that are used in a DSS from the applications that use them. The purpose of an MBMS is to transform data from the DBMS into information that is useful in decision making. Because many problems that the user of a DSS will cope with may be unstructured, the MBMS should also be capable of assisting the user in model building.
- *Dialog generation and management system* (*DGMS*). The main product of an interaction with a DSS is insight. Because their users are often managers who are not computer trained, DSSs need to be equipped with intuitive and easy-to-use interfaces. These interfaces aid in model building, but also in interaction with the model, such as gaining insight and recommendations from it. The primary responsibility of a DGMS is to enhance the ability of the system user to use and benefit from the DSS. In the remainder of this entry, we use the broader term user interface rather than DGMS.

Although various DSSs exists, the above three components can be found in many DSS architectures and play a prominent role in their structure. Interaction among them is shown in Figure 33.1.

Essentially, the user interacts with the DSS through the DGMS. This communicates with the DBMS and MBMS, which screen the user and the user interface from the physical details of the model base and database implementation.

NORMATIVE SYSTEMS

Normative and Descriptive Approaches

Whether one trusts the quality of human intuitive reasoning strategies has a profound impact on one's view of the philosophical and technical foundations of DSSs. There are two distinct approaches to supporting decision making. The first aims at building support procedures or systems that imitate human experts. The most prominent member of this class of DSSs are *expert systems*, computer

© 2012 Taylor & Francis Group, LLC

programs based on rules elicited from human domain experts that imitate reasoning of a human expert in a given domain. Expert systems are often capable of supporting decision making in that domain at a level comparable to human experts. Although they are flexible and often able to address complex decision problems, they are based on intuitive human reasoning and lack soundness and formal guarantees with respect to the theoretical reliability of their results. The danger of the expert system approach, increasingly appreciated by DSS builders, is that along with imitating human thinking and its efficient heuristic principles, we may also imitate its undesirable flaws.[4]

The second approach is based on the assumption that the most reliable method of dealing with complex decisions is through a small set of normatively sound principles of how decisions should be made. Although heuristic methods and ad hoc reasoning schemes that imitate human cognition may in many domains perform well, most decision makers will be reluctant to rely on them whenever the cost of making an error is high. To give an extreme example, few people would choose to fly airplanes built using heuristic principles over airplanes built using the laws of aerodynamics enhanced with probabilistic reliability analysis. Application of formal methods in DSSs makes these systems philosophically distinct from those based on ad hoc heuristic artificial intelligence methods, such as rule-based systems. The goal of a DSS, according to this view, is to support unaided human intuition, just as the goal of using a calculator is to aid human's limited capacity for mental arithmetic.

Decision-Analytic DSSs

An emergent class of DSSs known as *decision-analytic DSSs* applies the principles of decision theory, probability theory, and decision analysis to their decision models. Decision theory is an axiomatic theory of decision making that is built on a small set of axioms of rational decision making. It expresses uncertainty in terms of probabilities and preferences in terms of utilities. These are combined using the operation of mathematical expectation. The attractiveness of probability theory, as a formalism for handling uncertainty in DSSs, lies in its soundness and its guarantees concerning long-term performance. Probability theory is often viewed as the gold standard for rationality in reasoning under uncertainty. Following its axioms offers protection from some elementary inconsistencies. Their violation, however, can be demonstrated to lead to sure losses.[5] Decision analysis is the art and science of applying decision theory to real-world problems. It includes a wealth of techniques for model construction, such as methods for elicitation of model structure and probability distributions that allow minimization of human bias, methods for checking the sensitivity of a model to imprecision in the data, computing the value of obtaining additional information, and presentation of results (see, e.g., von Winterfeldt[6] for a basic review of the available techniques). These methods have been under continuous scrutiny by psychologists working in the domain of behavioral decision theory and have proven to cope reasonably well with the dangers related to human judgmental biases.

Normative systems are usually based on graphical probabilistic models, which are representations of the joint probability distribution over a model's variables in terms of directed graphs. Directed graphs, such as the one in Figure 33.2, are known as Bayesian networks (BNs) or causal networks.[7] Bayesian networks offer a compact representation of joint probability distributions and are capable of practical representation of large models, consisting of tens or hundreds of variables. Bayesian networks can be easily extended with decision and value variables for modeling decision problems. The former denote variables that are under the decision maker's control and can be directly manipulated, and the latter encode users' preferences over various outcomes of the decision process. Such amended graphs are known as *influence diagrams*.[8] Both the structure and the numerical probability distributions in a BN can be elicited from a human expert and are a reflection of the expert's subjective view of a real-world system. If available, scientific knowledge about the system, both in terms of the structure and frequency data, can be easily incorporated in the model. Once a model has been created, it is optimized using formal decision-theoretic algorithms. Decision analysis is based on the empirically tested paradigm that people are able to reliably store

© 2012 Taylor & Francis Group, LLC

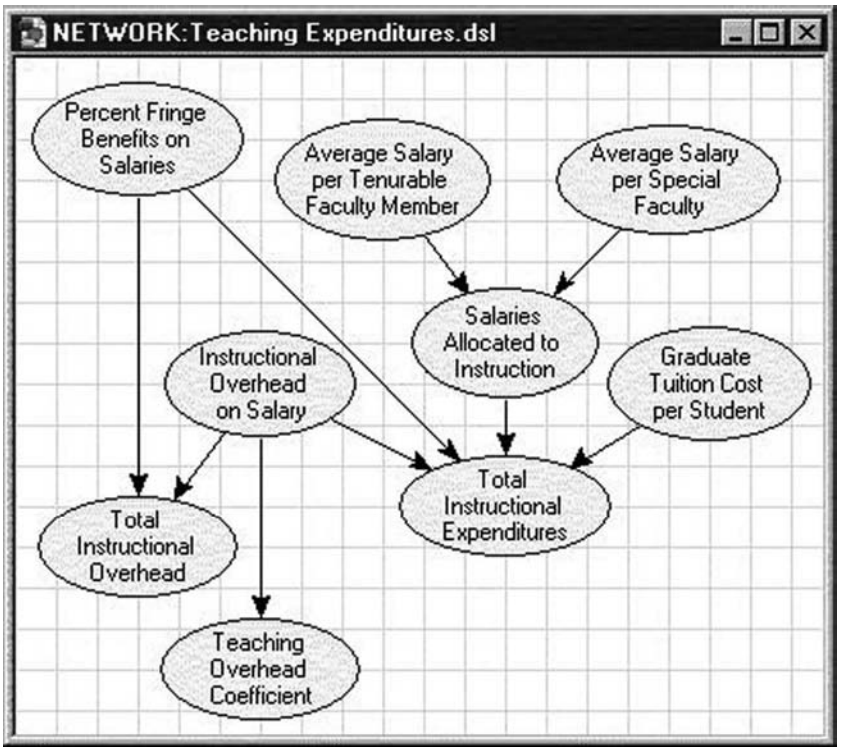

FIGURE 33.2 Example of a BN modeling teaching expenditures in university operations.

and retrieve their personal beliefs about uncertainty and preferences for different outcomes, but are much less reliable in aggregating these fragments into a global inference. Although human experts are excellent in structuring a problem, determining the components that are relevant to it and providing local estimates of probabilities and preferences, they are not reliable in combining many simple factors into an optimal decision. The role of a decision-analytic DSS is to support them in their weaknesses using the formal and theoretically sound principles of statistics.

The approach taken by decision analysis is compatible with that of DSSs. The goal of decision analysis is to provide insight into a decision. This insight, consisting of the analysis of all relevant factors, their uncertainty, and the critical nature of some assumptions, is even more important than the actual recommendation.

Decision-analytic DSSs have been successfully applied to practical systems in medicine, business, and engineering. Some examples of applications are described in a special issue of *Communications of the ACM* on practical applications of decision-theoretic methods (Vol. 38, No. 3, March 1995). We encourage the readers to experiment with GeNIe,[9] a development system for decision-analytic DSSs developed at the Decision Systems Laboratory, University of Pittsburgh, available at http://genie.sis.pitt.edu/. As these systems tend to naturally evolve into three not necessarily distinct classes, it may be interesting to compare their structure and architectural organization.

- *Systems with static domain models*. In this class of systems, a probabilistic domain is represented by a typically large network encoding the domain's structure and its numerical parameters. The network comprising the domain model is normally built by decision analysts and domain experts. An example might be a medical diagnostic system covering a certain class of disorders. Queries in such a system are answered by assigning values to those nodes of the network that constitute the observations for a particular case and

© 2012 Taylor & Francis Group, LLC

propagating the impact of the observation through the network to find the probability distribution of some selected nodes of interest (e.g., nodes that represent diseases). Such a network can, on a case-by-case basis, be extended with decision nodes and value nodes to support decisions. Systems with static domain models are conceptually similar to rule-based expert systems covering an area of expertise.

- *Systems with customized decision models*. The main idea behind this approach is automatic generation of a graphical decision model on a per-case basis in an interactive effort between the DSS and the decision maker. The DSS has domain expertise in a certain area and plays the role of a decision analyst. During this interaction, the program creates a customized influence diagram, which is later used for generating advice. The main motivation for this approach is the premise that every decision is unique and needs to be looked at individually; an influence diagram needs to be tailored to individual needs.[10]
- *Systems capable of learning a model from data*. The third class of systems employs computer-intensive statistical methods for learning models from data.[11–15] Whenever there are sufficient data available, the systems can literally learn a graphical model from these data. This model can be subsequently used to support decisions within the same domain.

The first two approaches are suited for slightly different applications. The customized model generation approach is an attempt to automate the most laborious part of decision making, structuring a problem, so far done with significant assistance from trained decision analysts. A session with the program that assists the decision maker in building an influence diagram is laborious. This makes the customized model generation approach particularly suitable for decision problems that are infrequent and serious enough to be treated individually. Because in the static domain model approach, an existing domain model needs to be customized by the case data only, the decision-making cycle is rather short. This makes it particularly suitable for those decisions that are highly repetitive and need to be made under time constraints.

A practical system can combine the three approaches. A static domain model can be slightly customized for a case that needs individual treatment. Once completed, a customized model can be blended into the large static model. Learning systems can support both the static and the customized model approach. However, the learning process can be greatly enhanced by prior knowledge from domain experts or by a prior model.

Equation-Based and Mixed Systems

In many business and engineering problems, interactions among model variables can be described by equations that, when solved simultaneously, can be used to predict the effect of decisions on the system, and hence support decision making. One special type of simultaneous equation model is known as the structural equation model (SEM), which has been a popular method of representing systems in econometrics. An equation is structural if it describes a unique, independent causal mechanism acting in the system. Structural equations are based on expert knowledge of the system combined with theoretical considerations. Structural equations allow for a natural, modular description of a system—each equation represents its individual component, a separable and independent mechanism acting in the system—yet, the main advantage of having a structural model is, as explicated by Simon,[16] that it includes causal information and aids predictions of the effects of external interventions. In addition, the causal structure of a SEM can be represented graphically,[16] which allows for combining them with decision-analytic graphical models in practical systems.[16,17]

Structural equation models offer significant advantages for policy making. Often a decision maker confronted with a complex system needs to decide not only the values of policy variables, but also which variables should be manipulated. A change in the set of policy variables has a profound impact on the structure of the problem and on how their values will propagate through the system.

© 2012 Taylor & Francis Group, LLC

The user chooses which variables are policy variables and which are determined within the model. A change in the SEMs or the set of policy variables can be reflected by a rapid restructuring of the model and predictions involving this new structure.[18]

Our long-term project, the Environment for Strategic Planning (ESP),[19] is based on a hybrid graphical modeling tool that combines SEMs with decision-analytic principles. The ESP is capable of representing both discrete and continuous variables involved in deterministic and probabilistic relationships. The powerful features of SEMs allow the ESP to act as a graphical spreadsheet integrating numerical and symbolic methods, and allowing the independent variables to be selected at will without having to reformulate the model each time. This provides an immense flexibility that is not afforded by ordinary spreadsheets in evaluating alternate policy options.

USER INTERFACES TO DSSs

Although the quality and reliability of modeling tools and the internal architectures of DSSs are important, the most crucial aspect of DSSs is, by far, their user interface. Systems with user interfaces that are cumbersome or unclear or that require unusual skills are rarely useful and accepted in practice. The most important result of a session with a DSS is insight into the decision problem. In addition, when the system is based on normative principles, it can play a tutoring role; one might hope that users will learn the domain model and how to reason with it over time, and improve their own thinking.

A good user interface to DSSs should support model construction and model analysis, reasoning about the problem structure in addition to numerical calculations, and both choice and optimization of decision variables. We discuss these in the following sections.

Support for Model Construction and Model Analysis

User interface is the vehicle for both model construction (or model choice) and for investigating the results. Even if a system is based on a theoretically sound reasoning scheme, its recommendations will only be as good as the model on which they are based. Furthermore, even if the model is a very good approximation of reality and its recommendations are correct, they will not be followed if they are not understood. Without understanding, the users may accept or reject a system's advice for the wrong reasons and the combined decision-making performance may deteriorate even below unaided performance.[20] A good user interface should make the model on which the system's reasoning is based transparent to the user.

Modeling is rarely a one-shot process, and good models are usually refined and enhanced as their users gather practical experiences with the system recommendations. It is important to strike a careful balance between precision and modeling efforts; some parts of a model need to be very precise, whereas others do not. A good user interface should include tools for examining the model and identifying its most sensitive parts, which can be subsequently elaborated on. Systems employed in practice will need their models refined, and a good user interface should make it easy to access, examine, and refine its models. Some pointers to work on support for building decision-analytic systems can be found in.[21–24]

Support for Reasoning about the Problem Structure in Addition to Numerical Calculations

Although numerical calculations are important in decision support, reasoning about the problem structure is even more important. Often when the system and its model are complex, it is insightful for the decision maker to realize how the system variables are interrelated. This is helpful not only in designing creative decision options, but also in understanding how a policy decision will affect the objective.

© 2012 Taylor & Francis Group, LLC

Graphical models, such as those used in decision analysis or in equation-based and hybrid systems, are particularly suitable for reasoning about structure. Under certain assumptions, a directed graphical model can be given a causal interpretation. This is especially convenient in situations where the DSS autonomically suggests decision options; given a causal interpretation of its model, it is capable of predicting effects of interventions. A causal graph facilitates building an effective user interface. The system can refer to causal interactions during its dialogue with the user, which is known to enhance user insight.[25]

Support for Both Choice and Optimization of Decision Variables

Many DSSs have an inflexible structure in the sense that the variables that will be manipulated are determined at the model-building stage. This is not very suitable for planning of the strategic type when the object of the decision-making process is identifying both the objectives and the methods of achieving them. For example, changing policy variables in a spreadsheet-based model often requires that the entire spreadsheet be rebuilt. If there is no support for that, few users will consider it as an option. This closes the world of possibilities for flexible reframing of a decision problem in the exploratory process of searching for opportunities. Support for both choice and optimization of decision variables should be an inherent part of DSSs.

Graphical Interface

Insight into a model can be increased greatly at the user interface level by a diagram representing the interactions among its components (e.g., a drawing of a graph on which a model is based, such as in Figure 33.2). This graph is a qualitative, structural explanation of how information flows from the independent variables to the dependent variables of interest. Because models may become very large, it is convenient to structure them into submodels, groups of variables that form a subsystem of the modeled system.[2,6] Such submodels can be again shown graphically with interactions among them, increasing simplicity and clarity of the interface. Figure 33.3 shows a submodel-level view of a model developed in our ESP project. Note that the graph in Figure 33.2 is an expanded version of the *Teaching Expenditures* submodel in Figure 33.3. The user can navigate through the hierarchy of

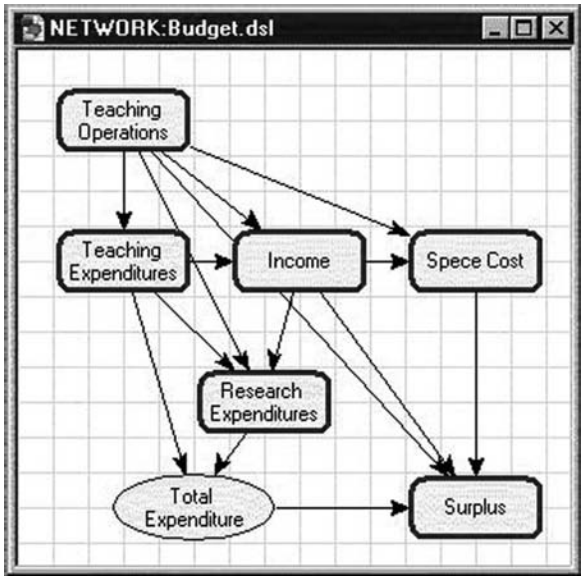

FIGURE 33.3 A submodel-level view of a decision model.

© 2012 Taylor & Francis Group, LLC

the entire model in her quest for insight, opening and closing submodels on demand. Some pointers to work on user interfaces of decision-analytic systems can be found in Wang,[24] Druzdzel[26,27] and Wiecha.[28]

CONCLUSION

Decision support systems are powerful tools integrating scientific methods for supporting complex decisions with techniques developed in information science and are gaining an increased popularity in many domains. They are especially valuable in situations in which the amount of available information is prohibitive for the intuition of an unaided human decision maker, and in which precision and optimality are of importance. Decision support systems aid human cognitive deficiencies by integrating various sources of information, providing intelligent access to relevant knowledge, aiding the process of structuring, and optimizing decisions.

Normative DSSs offer a theoretically correct and appealing way of handling uncertainty and preferences in decision problems. They are based on carefully studied empirical principles underlying the discipline of decision analysis, and they have been successfully applied in many practical systems. We believe that they offer several attractive features that are likely to prevail in the long run as far as the technical developments are concerned.

Because DSSs do not replace humans but rather augment their limited capacity to deal with complex problems, their user interfaces are critical. The user interface determines whether a DSS will be used at all and, if so, whether the ultimate quality of decisions will be higher than that of an unaided decision maker.

ACKNOWLEDGMENTS

Work on this entry was supported by the National Science Foundation under Faculty Early Career Development (CAREER) Program, grant IRI-9624629, by the Air Force Office of Scientific Research under grants F49620-97-1-0225, F49620-00-1-0112, and FA9550-06-1-0243 and by the University of Pittsburgh Central Research Development Fund. Figures 33.2 and 33.3 are snapshots of GeNIe, a general purpose development environment for graphical DSSs developed by the Decision Systems Laboratory, University of Pittsburgh, and available at http://genie.sis.pitt.edu/. We want to thank Ms. Nanette Yurcik for her assistance with technical editing.

REFERENCES

1. Kahneman, D.; Slovic, P.; Tversky, A., Eds. *Judgment Under Uncertainty: Heuristics and Biases*; Cambridge University Press: Cambridge, 1982.
2. Dawes, R.M. *Rational Choice in an Uncertain World*; Hartcourt Brace Jovanovich: San Diego, CA, 1988.
3. Sage, A.P. *Decision Support Systems Engineering*; John Wiley & Sons, Inc.: New York, 1991.
4. Henrion, M.; Breese, J.S.; Horvitz, E.J. Decision analysis and expert systems. AI Mag. Winter **1991**, *12* (4), 64–91.
5. Savage, L.J. *The Foundations of Statistics*, 2nd revised Ed.; Dover Publications: New York, 1972.
6. von Winterfeldt, D.; Edwards, W. *Decision Analysis and Behavioral Research*; Cambridge University Press: Cambridge, 1988.
7. Pearl, J. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*; Morgan Kaufmann Publishers, Inc.: San Mateo, CA, 1988.
8. Howard, R.A.; Matheson, J.E. Influence diagrams. In *The Principles and Applications of Decision Analysis*; Howard, R.A., Matheson, J.E., Eds.; Strategic Decisions Group: Menlo Park, CA, 1984; 719–762.
9. Druzdzel, M.J. SMILE: Structural Modeling, Inference, and Learning Engine and GeNIe: A development environment for graphical decision-theoretic models. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence (AAAI-99)*, Orlando, FL, 1999; 902–903.

© 2012 Taylor & Francis Group, LLC

10. Holtzman, S. *Intelligent Decision Systems*; Addison-Wesley: Reading, MA, 1989.
11. Spirtes, P.; Glymour, C.; Scheines, R. *Causation, Prediction, and Search*; Springer Verlag: New York, 1993.
12. Pearl, J.; Verma, T.S. A theory of inferred causation. In *KR-91, Principles of Knowledge Representation and Reasoning*, Proceedings of the Second International Conference, Cambridge, MA, Allen, J.A., Fikes, R., Sandewall, E., Eds.; Morgan Kaufmann Publishers, Inc.: San Mateo, CA, 1991; 441–452.
13. Cooper, G.F.; Herskovits, E. A Bayesian method for the induction of probabilistic networks from data. Mach. Learn. **1992**, *9* (4), 309–347.
14. Glymour, C.; Cooper, G.F., Eds. *Computation, Causation, and Discovery*; AAAI Press: Menlo Park, CA, 1999.
15. Heckerman, D.E.; Geiger, D.; Chickering, D.M. Learning Bayesian networks: The combination of knowledge and statistical data. Mach. Learn. **1995**, *20* (3), 197–243.
16. Simon, H.A. Causal ordering and identifiability. In *Studies in Econometric Method. Cowles Commission for Research in Economics*; Monograph No. 14; Hood, W.C., Koopmans, T.C., Eds.; John Wiley & Sons, Inc.: New York, 1953; Chapter III, 49–74.
17. Druzdzel, M.J.; Simon, H.A. Causality in Bayesian belief networks. In *Proceedings of the Ninth Annual Conference on Uncertainty in Artificial Intelligence (UAI-93)*; Morgan Kaufmann Publishers, Inc.: San Francisco, 1993; 3–11.
18. Simon, H.A.; Kalagnanam, J.R.; Druzdzel, M.J. Performance budget planning: The case of a research university; 2000. Unpublished manuscript.
19. Druzdzel, M.J. ESP: A mixed initiative decision-theoretic decision modeling system. In *Working Notes of the AAAI-99 Workshop on Mixed-Initiative Intelligence*; Orlando, FL, 1999; 99–106.
20. Lehner, P.E.; Mullin, T.M.; Cohen, M.S. A probability analysis of the usefulness of decision aids. In *Uncertainty in Artificial Intelligence 5*, Henrion, M., Shachter, R.D., Kanal, L.N., Lemmer, J.F., Eds.; Elsevier Science Publishers: B.V.: North Holland, 1990; 427–436.
21. Druzdzel, M.J.; Díez, F.J. Criteria for combining knowledge from different sources in probabilistic models. J. Mach. Learn. Res. **2003**, *4* (July), 295–316.
22. Druzdzel, M.J.; van der Gaag, L.C. Building probabilistic networks: "Where do the numbers come from?" guest editors' introduction. IEEE Trans. Knowl. Data Eng. **2000**, *12* (4), 481–486.
23. Lu, T.-C.; Druzdzel, M.J. Causal mechanism-based model construction. In *Proceedings of the Sixteenth Annual Conference on Uncertainty in Artificial Intelligence (UAI-2000)*; Morgan Kaufmann Publishers, Inc.: San Francisco, 2000; 353–362.
24. Wang, H.; Druzdzel, M.J. User interface tools for navigation in conditional probability tables and elicitation of probabilities in Bayesian networks. In *Proceedings of the Sixteenth Annual Conference on Uncertainty in Artificial Intelligence (UAI-2000)*; Morgan Kaufmann Publishers, Inc.: San Francisco, 2000; 617–625.
25. Druzdzel, M.J. Probabilistic reasoning in decision support systems: From computation to common sense. Ph.D. thesis, Department of Engineering and Public Policy, Carnegie Mellon University: Pittsburgh, PA, 1992.
26. Druzdzel, M.J. Five useful properties of probabilistic knowledge representations from the point of view of intelligent systems. Fundamenta Informaticæ **1997**, *30* (3/4), 241–254. (special issue on knowledge representation and machine learning).
27. Druzdzel, M.J. Explanation in probabilistic systems: Is it feasible? Will it work? In *Proceedings of the Fifth International Workshop on Intelligent Information Systems (WIS-96)*, Deblin, Poland, 1996; 12–24.
28. Wiecha, C.F. An empirical study of how visual programming aids in comprehending quantitative policy models (Volumes I and II), Ph.D. thesis, Department of Engineering and Public Policy, Carnegie Mellon University: Pittsburgh, PA, 1986.

© 2012 Taylor & Francis Group, LLC

35 Geographic Information Systems (GIS)

Timothy F. Leslie and Nigel M. Waters

CONTENTS

© 2012 Taylor & Francis Group, LLC

INTRODUCTION AND OVERVIEW

In 1998, *The Encyclopedia of Library and Information Science* published an entry on geographic information systems (GIS)[1] that reviewed both the history and the body of knowledge associated with GIS. A sequel in 2001[2] documented the progress made by the community during the following 3 years, with particular detail given to the "systems versus science" debate. For the sake of completeness and self-containment, this overview of the subject again begins with formal definitions of GIS. Next, the state of the art as it existed in 2008 is described. This is followed by a discussion of spatial thinking as conceptualized within the (generally American) educational system.

In recent years, GIS has come to represent a synthesis of science and application. The "systems versus science" debate has become passé. Internet applications have flourished, with many users unaware they were using GIS technology to create maps or obtain driving directions. This entry concludes with an overview of the near-term future of GIS and with a list of GIS resources, both online and traditional print materials. The present account does not provide a complete and comprehensive introduction to GIS, and readers wishing to learn the basics before consulting the rest of the entry are advised to go to the following online tutorials: the U.S. Geological Survey (USGS) GIS education Web site at http://education.usgs.gov/common/lessons/gis.html and the Environmental Systems Research Institute (ESRI) discussion of GIS presented at http://www.gis.com/whatisgis/. A comprehensive description of those topics belonging to the body of knowledge associated with GIS may be found in DiBiase et al.[3]

In addition, this entry does not review the history of GIS. The reader may consult the extensive discussion in Waters[1] or in Clarke.[4] A complete review may be found in Foresman[5] and comprehensive online resources are maintained with The GIS History Project (http://www.ncgia.buffalo.edu/gishist/). Chrisman[6] has described the transformation of computer mapping software into GIS at the Harvard Laboratory for Computer Graphics and Spatial Analysis during the 1960s and 1970s.

MODERN DEFINITIONS OF GIS

A terse, useful definition of GIS continues to elude the community. Two views of GIS pervade the literature, differing largely because of the difference in the "S" in the acronym. Those scholars that represent the "S" as systems include Clarke,[4] who provides a number of definitions of GIS. Clarke begins by stating that a GIS is a computer-based system for linking attribute data from a database with spatial information. He notes that a GIS can be described in various ways. Thus some authors have referred to GIS as a toolbox. Similarly, Burrough and McDonnell[7] state that GIS is a "a powerful set of tools for storing and retrieving at will, transforming and displaying spatial data from the real world for a particular set of purposes." Longley et al.[8] review definitions that describe GIS as both data analysis–data display tools and as map-making tools. These definitions emphasize the applied nature of GIS and are generally used by practitioners in the field, such as the government and related industry contractors. These definitions have become more entrenched with the increasing use of software programming packages and languages (e.g., Visual Basic, Python, and Java, among others) to create sets of procedures that specialized user groups can employ (e.g., transportation planners, see Kang and Scott[9]).

Alternatively, the "S" in GIS can be taken to represent Science[10]. This approach has been advocated by scholars who are actively developing new methods and who view themselves as more than simple toolmakers. Goodchild provides an overview of the differences between GISystems, GIScience and GIStudies at http://www.ncgia.ucsb.edu/giscc/units/u002/u002.html. According to Goodchild GIScience is the science behind the technology of GIS. It is also the science that keeps GIS at the research frontier. GIScience is thus a multidisciplinary field in which cartography, cognitive psychology, computer science, geodesy, geography, photogrammetry, and spatial statistics are all important contributors.

The tool versus science debate has been reviewed by Wright et al.[11] It has been resolved largely by the acceptance of both terms and an increased vagueness in the use of the GIS acronym. Within

© 2012 Taylor & Francis Group, LLC

universities this dichotomy is evident in the number of "professional master's" programs available largely to fill the market for increased application courses and community-based GIS funding in the vein of GISystems. GIScience remains as a realm for continued research and software development, and is popular as a specialization, minor, or additional certificate in degree programs. Academic units with a mix of GIScience and GISystems activity remain healthy.

Finally, Chrisman[12] has defined GIS as an "organized activity by which people measure and represent geographic phenomena then transform these representations into other forms while interacting with social structures." This definition reflects the increased interest in the use of GIS for community planning and advocacy. It is such an important new trend that it has been variously referred to as community-based GIS and Public Participation GIS (PPGIS) and more recently as VGIS where the "V" in the acronym indicates volunteer involvement. These developments are described in further detail below.

GIS: THE STATE OF THE ART IN 2008

In 2008, GIS software packages for making maps and for displaying and analyzing spatial data in a variety of ways was commonplace. Large price differences existed, with GIS software packages ranging in cost from free (for GRASS and other open source initiatives) to a few hundred dollars (for Idrisi, MapInfo, and Maptitude) to tens of thousands of dollars (for enterprise versions of TransCAD and ArcServer). These packages generally come with a graphical interface and run on the Windows operating system, although Unix-friendly server editions are becoming common. Mac OS X and Linux are poorly represented, and can only run a subset of existing GIS software without a Windows emulator or interpreter. Open source software has been particularly successful with these operating systems, to the point that dedicated teams focus on GIS-specific Linux distributions (see information on DebianGIS at http://wiki.debian.org/DebianGis).

Conducting analysis with GIS software still requires extensive training and this is especially so if it is to be used for decision-making and policy implementation. Most GIS education and training is completed in university undergraduate programs. Post-degree diploma programs are also popular as are graduate level master's degree programs, and employers frequently pay for such education for their employees.

Web-based GIS applications and the use of software and data online are becoming increasingly common. Many of these Web-based devices are lowering the technical know-how necessary to interact with spatial data. GPS units are capable of calculating driving directions as well as tracking traffic information from a server and rerouting the user on the fly. Cell phones, such as the iPhone, can track their location, navigate users, and check the weather with a few touches of the screen.

Spatial data is still extremely costly in most countries where cost-recovery models are often used by government agencies (see Taylor[13] for an exhaustive discussion of this topic for various countries around the world). The United States is almost the lone exception to this approach to the provision of spatial data, and it is arguable that this has done much to spawn the world's most active and innovative GIS industry.

Although GIS, even today, cannot be considered more than a niche application it is now a common place subject in university curricula and is frequently used as a research tool by a large number of university disciplines.[1] In addition, it is being taught more and more in the K-12 curriculum in schools and is being used in an increasingly extensive number of applications in both the public and private sectors.

Public Participation GIS and Volunteered Geographic Information

During the last decade, GIS has been used more and more for community planning and social advocacy. Such developments have been variously described as Public Participation GIS and Participatory GIS with the acronyms PPGIS and PGIS, respectively, in common use. The most extensive set of

© 2012 Taylor & Francis Group, LLC

resources for participatory GIS may be found at the portal Web site maintained by the integrated approaches to participatory development (IAPAD) organization at http://www.iapad.org/. IAPAD maintains a list for those interested in PGIS research and also stores numerous case studies which may be downloaded. In recent years it has promoted as participatory three-dimensional modeling (P3DM) of physical environments and the ethically responsible use of GIS to protect lands belonging to indigenous communities.

PGIS has now been well accepted by mainstream GIS researchers with highly regarded texts such as that by Craig et al.[14] devoted to this topic. For a number of years PGIS had its own series of conferences sponsored by the Urban and Regional Information Systems Association (URISA) although during 2006 and 2007 PGIS was again merged into URISA's main, annual conference. The PPGIS Web site (http://www.ppgis.net/) maintains an open forum on Participatory GIS and associated technologies.

Volunteered geographic information is an increasingly important and associated development. Software developments that include Google Earth, Google SketchUp, Wikimapia, and OpenStreetMap have allowed citizens with limited or indeed no specialized knowledge of GIS to upload their geographic knowledge to publicly accessible Web sites. This process of "geotagging" and its impact on the future of GIS is discussed by Goodchild.[15] The Geography Network (http://www.geographynetwork.com/) supports project Globe (http://www.globe.gov/GaN/analyze.html) which allows students in elementary schools to observe data, for example, the brilliance of the night sky. It is easy for these students to record their observations and upload them to a map where they can become part of a network of thousands of observations from schoolchildren around the world. As Goodchild notes, the children have become geographic sensors.

TEACHING SPATIAL THINKING

GIS Education continues to progress as spatial thinking has received attention at all educational levels. Many vendors of GIS software offer reduced or free versions of their packages for education institutions, and resource materials including data sets and lesson plans are widely available.

Schools

Recently the National Research Council has produced a major study[16] advocating the teaching of spatial thinking and GIS across the K-12 curriculum. The authors of the report argue that spatial thinking is a constructive mix of three elements: spatial concepts, methods of representation, and spatial reasoning. Indeed the Association of American Geographers has argued recently (http://aag.org/nclb/nclb.pdf) for changes to the U.S. No Child Left Behind Legislation that would see an appropriation of funding in this legislation for the teaching of geography and GIS.

It can be argued that GIS should be incorporated into the K-12 curriculum for several reasons. First, it helps with the teaching of geography, a core academic discipline. Major software manufacturers such as ERSI (http://www.esri.com) have made available at no cost software such as ArcGIS Explorer which, at the time of writing is available with seven worldwide coverages that include various themes such as physical relief and political boundaries. Second, spatial thinking is advocated because it helps with other disciplines such as the physical, mathematical and environmental sciences. Third, it prepares students to be better citizens in that the data embedded within a GIS provides them with an understanding of other regions of their country and of other countries within the world. A GIS also prepares them to interact with the world in a more effective manner as an entrepreneur or merely as someone who can use an in-car navigation system more resourcefully.

Evidence to support improved spatial thinking and education in the National Research Council Report is contained in Chapter 4 and Appendix C of the study. Unfortunately, most of this research

© 2012 Taylor & Francis Group, LLC

is dated and will have to be revisited if the council is to succeed in its goal of developing new GIS software that is age appropriate in its design, scope, and sequence.

Information on geographic information technology for teachers and the lay person may be found at http://geography.about.com/od/geographictechnology/Geographic_Technology.htm. A complete set of links summarizing articles, lesson plans, and software for teaching GIS in the K-12 curriculum is available at http://www.esi.utexas.edu/gk12/lessons.php.

GIS and geography teaching in elementary and secondary schools has moved forward quickly since 1990. Bednarz and Bednarz[17] take an optimistic view of the progress that has been made and how future challenges may be addressed. Doering (http://gis2.esri.com/library/userconf/educ02/pap5039/p5039.htm) has analyzed the effectiveness of various strategies for teaching about GIS in the K-12 curriculum (see also Doering and Veletsianos[18]).

Simply put, GIS is a highly effective way of teaching schoolchildren about their world. There is, however, a steep learning curve for teachers and professional development resources constantly need to be upgraded (McClurg and Buss[19]). Others have argued for a minimal GIS software package that increases in complexity with grade level and focuses on the introduction of geographical concepts appropriate to a child's intellectual development (Marsh et al.[20]).

Resources for teachers may be found at a link on the ESRI Web site at http://www.esri.com/industries/k-12/education/teaching.html. These resources include lesson plans for a variety of ages and skill levels A list of resources for teachers including annotated bibliographies of the use of GIS in the K-12 system may be found at the Web site http://gislounge.com/k-12-education-in-gis/. Links to resources on best practices and "white papers" discussing the future of GIS in school education may be found at this link on the ESRI Web site: http://www.esri.com/library/whitepapers/pdfs/higher_ed.pdf.

The work of the National Center for Geographic Information and Analysis (NCGIA) at the University of California at Santa Barbara in supporting the integration of GIS into the secondary school curriculum may be seen at the following Web site: http://www.ncgia.ucsb.edu/education/projects/SEP/sep.html. This Web site also contains links to other sites providing resources and support for K-12 GIS initiatives. Resources for schools in the United Kingdom and a sourcebook that may be ordered online can both be found at http://www.abdn.ac.uk/gis_school/.

A new trend is the linking of qualitative geography to GIS (Mei-Po Kwan[21]). This development may also unite interest in another new area of research, Children's Geographies (see the new journal of that name and introductory editorial by Matthews[22]). Children and youths may be used to supply volunteer information that can be incorporated into GIS (see discussion above and Dennis[23]).

Despite all these developments the reality is that in the year 2008 many schools still do not have the computers or the teacher expertise to take advantage of the resources that are available to them on the Internet. It can only be hoped that this will change in the coming years.

Universities

University education in GIS grew substantially after the introduction of the core curriculum in GIS by the NCGIA in 1990. The original core curriculum was designed to provide university faculty with notes for 75 lectures that represented a year-long introduction to the fundamental issues and concepts in GIS. This curriculum was remarkably successful and about 2000 copies were distributed to over 70 countries after being translated into at least eight languages (including Portuguese, Chinese, Hungarian, Japanese, Korean, Polish, Russian, and French). It may still be found at http://www.geog.ubc.ca/courses/klink/gis.notes/ncgia/toc.html.

The new Core Curriculum in GIScience may be found at http://www.ncgia.ucsb.edu/education/curricula/giscc/ and is still under development. It includes two sets of lecture notes specifically on teaching GIS within a university setting http://www.ncgia.ucsb.edu/education/curricula/giscc/units/

© 2012 Taylor & Francis Group, LLC

u158/u158_f.html. A core curriculum for the closely related field of remote sensing may be found at http://userpages.umbc.edu/~tbenja1/umbc7/.

A related occurrence has been the NCGIA's development of CSISS (The Center for Spatially Integrated Social Science http://www.csiss.org/index.html).

GIS research and teaching in Universities in the United States has been substantially stimulated through the creation of the University Consortium for Geographic Information Science [UCGIS; (http://www.ucgis.org/)]. The UCGIS defines its mission to be "an effective, unified voice for the geographic information science research community." A listing of university-based, GIS courses in the United Kingdom may be found at http://www.agi.org.uk/ under the Education Link. University based GIS research in the U.K. was also supported by the Regional Research Laboratory initiative.[1] Canadian GIS degree programs may be accessed at http://www.canadian-universities.net/Universities/Programs/Geography_and_GIS.html.

Masters Courses

In recent years, master's degrees have proliferated at universities in the United States and in many other countries around the world. A listing of these programs, including distance-based offerings, may be found at http://www.ucgis.org/priorities/education/GIS_Cert+Masters_Prog/certificates.htm. Many of these master's degree programs now include modules on programming in GIS. Popular choices for programming languages include Visual Basic, Java, C, C#, and C++. Students find these courses most attractive and often feel that their education in the GISciences is not complete without some basic training in programming. The more important software vendors such as ESRI (see below) are moving away from their own, proprietary scripting languages toward industry standard languages such as Visual Basic.

In some cases these masters programs have been seen as terminal, professional degree programs which supply a need generated by the GIS industry. Others have seen them as the ideal "springboard" into Ph.D. research in Geography and other disciplines such as Archaeology that use spatially distributed data (see the Web site at http://www.le.ac.uk/geography/postgraduate/msc_gis_hg.html which discusses the Master of Science degree in GIS at the University of Leicester).

Colleges

The NCGIA has developed a core curriculum for technical programs taught in colleges and this may be accessed at http://www.ncgia.ucsb.edu/education/curricula/cctp/Welcome.html. GIS has found a particularly successful niche in technical colleges that offer postgraduate diploma programs. One of the oldest and most successful of these programs has been taught at the College of Geographic Sciences in Nova Scotia, Canada, since the early 1980s. A description of this program may be found at http://www.cogs.ns.ca/Programs/Geomatics/. A partial listing of some of the better known college programs in GIS may be found at http://www.ncgia.ucsb.edu/education/curricula/cctp/resources/example_courses/examples_f.html.

Virtual Campuses

Distance education is a well-established method of instruction in GIS and is sponsored by the UCGIS organization among others. A "white paper" on this topic may be found at (http://dusk.geo.orst.edu/disted/). Links to many U.S. sites that offer distance education may be found at this location together with a link to the UNIGIS International site (http://www.unigis.org/) which has offices in 10 separate countries around the world. Perhaps one of the most outstandingly successful attempts at distance education is ESRI's virtual campus which may be found at http://training.esri.com/gateway/index.cfm?fa=trainingOptions.gateway. These courses may be either self study or instructor led.

While distance-based education represents an affordable and convenient way of learning about GIS or indeed any other subject it is not without its critics such as Noble.[24]

© 2012 Taylor & Francis Group, LLC

SOFTWARE PACKAGES

Software vendors have done much to popularize the use of GIS in academia, government, and industry. This they have achieved by sponsoring software distribution, conferences, Web sites, Web services, and trade newsletters. Here the activities of a number of the more important vendors and software developers are described. Most software vendors now support their own online listserves, Web knowledge banks, and other interactive communities in order to resolve problems for their user base. Information on Open Source GIS software may be found at the Open Source Geospatial Foundation Web site (http://www.osgeo.org/) and is discussed in more detail below. A survey of this software undertaken in late 2007 is available at http://www.foss4g2007.org/presentations/view.php?abstract_id=136. The rest of this section lists the leading commercial GIS software.

ESRI

Founded in 1969, ESRI (http://www.esri.com/) continues to dominate the industry as the GIS market leader. ESRI offers various configurations of its ArcGIS software. The current version of the ArcGIS software is 9.3 but new releases occur about every 6 months. The Desktop configuration has three components: ArcGIS Desktop, ArcGIS Engine, and ArcGIS Explorer (http://www.esri.com/products.html#arcgis). The Desktop product allows for the creation, editing, and analysis of geographic data and the development of professional, publication-quality maps. ESRI provides a server configuration for delivering maps and geographic data across an organization and over the Web. This configuration requires their ArcGIS Server and Image Server products. ESRI's Mobile GIS products include ArcGIS Mobile and ArcPad, products that allow the development of GIS products in the field and full use by clients with mobile devices including phones. ESRI offers data in various formats to populate these GIS products and also as Web services that are available online (http://www.esri.com/software/arcwebservices/index.html). Other organizations that offer Web services include GIS factory (http://gisfactory.com/webservices.html) where the services include address finders, district finders, and route finders (http://gisfactory.com/whitepapers/wp_giswebservices.pdf).

ESRI sponsors the ArcWatch e-mail newsletter, the ArcUser magazine, and the ArcNews publication. In 2008 it will hold its 28th annual user conference (http://www.esri.com/events/uc/index.html), one of the most popular and enduring of all the yearly GIS conferences. Recently attendance at this premier, vendor-sponsored conference has been around 14,000 attendees. The functionality of the ESRI ArcGIS software has been augmented by a series of extensions that can be deployed to perform specific functions. For ArcGIS these include extensions for analysis, such as Spatial Analyst and Network Analyst, for productivity including, Publisher and Street Map, and solution based software such as Business Analyst and Military Analyst and, finally, Web services. A complete list of ESRI supported extensions may be found at http://www.esri.com/software/arcgis/about/desktop_extensions.html. Extensions developed by their partners may be found at http://gis.esri.com/partners/partners-user/index.cfm. A review of these extensions, organized by application type, is provided by Limp,[25] an article which may be accessed by registering at the GeoPlace Web site (http://www.geoplace.com), a GIS industry Web portal. Some extensions are packaged in the form of toolboxes that perform specific GIS operations that are often missing from the standard GIS packages. A prototypical example is Hawth's Tools that provides functionality for a variety of spatial, sampling, and animal movement operations and may be found at the spatial ecology Web site (http://www.spatialecology.com/htools/tooldesc.php).

IDRISI

One of the most popular, affordably priced, GIS products is Idrisi which was developed in 1987 by Ron Eastman and is now supported by Clark Labs at Clark University in Worcester, Massachusetts

© 2012 Taylor & Francis Group, LLC

(http://www.clarklabs.org/). Idrisi's roots are as a raster GIS and as such it has been most widely used in resource management, land use change, and image processing applications. At the time of writing, the Andes Edition, the 15th major release, was the current version of this enormously popular GIS software package. The unusual name of the software owes its origins to the famed, twelfth century, Moroccan cartographer, Muhammad al-Idrisi. The Idrisi software is a fully functional GIS and image processing package that is now used in more than 175 countries. It has an especially rich and diverse set of processing modules for analytical research that include the first ever machine learning algorithms for use in a GIS and image processing system, soft classifiers, multi-criteria and multi-objective decision making that provided the first GIS implementation of Saaty's Analytical Hierarchy Process (Saaty[26]), sophisticated geospatial statistics, and a dynamic modeling routine that is implemented through a graphical interface.

Intergraph

Intergraph is ESRI's chief competitor for the title of GIS market leader and has been providing GIS and related software for 35 years. Intergraph has a suite of GIS-related products including its GeoMedia products (http://www.intergraph.com/geomediasuite/). Intergraph also sponsors its own annual user's conference and publications including the trade publication, Insight, which is available online together with Intergraph's e-Connection Newsletter. Intergraph works with business partners such as Hansen Information Technologies (http://www.hansen.com/) to provide additional geospatial functionality, in this case for asset management and transportation and related solutions.

MAPINFO

Since 1986 MapInfo Corporation, Troy, New York (http://www.mapinfo.com/) has been producing affordable GIS software that is eminently suited to desktop mapping and such applications as geodemographics and target marketing. MapInfo emphasizes location-based intelligence especially in the field of business planning and analysis. It too supports an annual conference, the MapWorld Global User Conference, and provides customer support through online user groups.

Caliper Corporation

Caliper Corporation, Newton, Massachusetts (http://www.caliper.com/), produces one of the most sophisticated low-cost GIS desktop mapping products available, Maptitude. This software comes complete with extensive data sets from the U.S. Bureau of the Census and is ideal software for many GIS applications and has been favorably reviewed. A special version of Maptitude is available for building and analyzing political and other redistricting plans. Caliper Corporation's flagship product is TransCAD, a transportation GIS package that has the most complete set of transportation planning and related routines available in any GIS package. The latest release of this software, Version 5, is also produced as a Web mapping package that may be used for developing online transportation planning applications. One suggestion is that this software could be used to do online travel surveys greatly reducing the cost of traditional in-house, paper-based surveys (http://www.caliper.com/web/gist2002.pdf). Caliper Corporation is now marketing a GIS-based traffic simulation package, TransModeler.

Autodesk

Autodesk, San Rafael, California (http://www.autodesk.com), is the major software developer in the Computer Assisted Drafting market with its AutoCAD product. In recent years it has also added desktop mapping and GIS to its product line with its Map 3D product.

© 2012 Taylor & Francis Group, LLC

Bentley

In a 2006 study the Daratech organization (http://www.daratech.com/) rated Bentley Systems, Inc., as the number two provider of GIS systems worldwide. Their flagship GIS/CAD product, Microstation, was originally developed for Intergraph. It is now available as Bentley Map.

Manifold

Manifold (http://www.manifold.net/index.shtml), manufactured by CDA International Ltd., is a low cost GIS that is highly popular with organizations that have limited budgets and lack the technical expertise to work with open source software. It has an online users' support group (http://forum.manifold.net/forum/). It is a full featured GIS that in its current release of 8.0 offers 64 bit processing, an Internet map server, and is available in personal and enterprise editions.

Free and Open Source Software

There are numerous GIS packages now available in various amounts of free and open-source packages. GRASS (the geographic resources analysis support system) has made large strides in development since its release under the open source GPL license in 1999 (http://grass.itc.it/). It is designed primarily to work on Linux and other operating systems that use X windows (not to be confused with Microsoft Windows).

GeoDA is a specialized analysis tool used to examine spatial autocorrelation and related spatial regression analyses implemented on Windows. PySal (python spatial analysis library) is a shared set of libraries for both GeoDA and the STARS software that is available at the Regional Analysis Laboratory at San Diego State University (http://regionalanalysislab.org/).

Software, such as the crime analysis package, Crimestat (http://www.icpsr.umich.edu/CRIMESTAT/), are free and used frequently in the professional world, although they are not truly open source. GIS also shares a great deal of overlap with the postgreSQL and mySQL server backends, and postgis serves as "spatially-enabled" upgrade for postgreSQL and has been implemented in both the U.S. and U.K. Programs such as terraview (http://www.terralib.org/), and mapserver (http://mapserver.gis.umn.edu/) have more niche audiences but are also growing in popularity.

Geoexploration Systems

There now exist a number of competing technologies that have been described as geographic exploration systems (Ball[27]) or geoexploration systems. These technologies include Google Earth, Microsoft's Virtual Earth, NASA WorldWind, and ESRI's ArcGIS Explorer among others. They have become extremely popular since the introduction of "mashup" technology that allows even the neophyte user to combine their spatial data with real world environments across a nation or indeed across the globe. Visualization software such as GeoFusion (http://www.geofusion.com/) has been developed to improve download times, allow the integration of multiple data sets, and enhance the interface of these systems. Geoexploration systems have proved extremely useful in aiding the development of participatory GIS where nonspecialists use GIS technology for advocacy planning or to protect the rights of indigenous populations (see discussion above). Volunteer geographic information has been made far more effective by the ease of use of this new type of GIS.

Geographic social networking is a new development that represents the integration of social network technology such as MySpace, video technology such as YouTube, and geoexploration systems like Virtual Earth. This approach is being pioneered by The Carbon Project (http://www.thecarbonproject.com/).

© 2012 Taylor & Francis Group, LLC

SPATIAL AUTOCORRELATION

Spatial analysis continues to be the crux of GIScience's growth. The forms of analysis special to geographic information have continued to be developed and remain unique to the discipline (Gould[28]). Spatial autocorrelation, the problem of observations near each other having correlated regression residuals, and related analysis has become ubiquitous in the geography literature. Increases in computational resources have allowed for most desktop computers to be able to create weight matrices, calculate spatial autocorrelation, and map significance scores (Anselin[29,30]; and Anselin and Florax[31]). These tools were originally implemented in stand-alone software, but are increasingly part of commercial software such as Idrisi and ArcMap.

For more sophisticated forms of analysis, researchers are still forced to use packages such as SpaceStat (http://www.spacestat.com/) or the spatial statistics routines in S-Plus (http://insightful.com). Modern spatial analysis continues to focus on local models of spatial association (Fotheringham et al.[32]). These Local Indicators of Spatial Association (LISA) statistics, such as Local Moran's I, are frequent in the literature. Anselin's GeoDA software is the most frequently used software employed to examine these local autocorrelation statistics. GeoDA allows for the creation and analysis of weight matrices, as well as the use of them to account for spatial autocorrelation in modified regression analysis. Another approach has been to allow the coefficients within regressions to vary over space. This method, termed geographically weighted regression, is promoted by Fotheringham and has received a mixed reception in the literature.

Markup Languages

Markup languages are the *lingua franca* of the Internet. Since its inception hypertext markup language (HTML) has been the dominant method for encoding information for text that is transmitted over the Internet. Essentially HTML does little more than provide a "picture" of a document for the Web user. All markup languages seek to provide information about the data that is transmitted over the Internet. When that data has unique characteristics, as is the case with spatial or geographical data, it requires its own markup language.

Geography markup language has been in development since 1998 and this has been largely due to the efforts of Ron Lake and his company Galdos Systems (http://www.galdosinc.com/). GML v3.0 was released as ISO Standard 19136 for the storage and transport of geographic data. GML is now the standard for the GeoWeb (http://www.geoweb.org/). It thus allows devices that are connected to the Internet to store and transmit geographical data across the Internet permitting the efficient use of Web services. Like XML, it has also spawned other related markup languages including CityGML which enables the storage and exchange of three dimensional objects that describe urban infrastructure (http://www.citygml.org/). In late 2007, CityGML was officially adopted by the Open Geospatial Consortium as the preferred markup language for urban infrastructure.

More commonly, GIS data on a server is accessed through flash and JavaScript applets that do not require the screen to refresh every time the user makes a change but instead the onscreen image will change "on the fly." This has vastly increased the usability of many online GIS applications. However, it also has made it far more difficult to create these GIS systems, with more advanced training required for these software and database packages.

GIS AND ITS APPLICATIONS

A major strength of GIS has been its ability to prove itself useful in a great many application areas. The reader may find detailed discussions of the use of GIS in the management of utilities, telecommunications, emergency management, land administration, urban planning, military applications, library management, health care, political redistricting, geodemographics and target marketing, agriculture and environmental monitoring in Longley et al.[8] Each of these application areas has an

© 2012 Taylor & Francis Group, LLC

extensive literature of its own and these are described in the various chapters included in Longley et al.'s comprehensive review of the discipline.

CERTIFICATION OF GIS PROFESSIONALS

An ongoing concern for GIS professionals has been the need for certification. Many individuals and organizations have argued that GIS professionals should be certified in a manner similar to the certification of engineers, geologists, psychologists, and others in professional disciplines. In 1998 the Urban and Regional Information Systems Association (URISA: http://www.urisa.org/) created a Certification Committee. After extensive industry-wide debate, the finalized portfolio-based certification program was established in 2003. This certification process was adopted and administered by the newly established GIS Certification Institute (http://www.gisci.org/). Certification involves establishing evidence of professional competence and ethical conduct. Until January 1, 2009, a "grandfathering" process was also permitted. At the end of 2007 almost 2000 individuals have availed themselves of the certification process.

GIS AND THE FUTURE

Judging the future of the discipline is difficult, as rapid advancements make such statements outdated by the time of publishing. Such is the case in a recent work by Reuter and Zipf (http://www.i3mainz.fh-mainz.de/publicat/zipf05/gis.where.next-reuter-zipf.pdf), that predicts the trajectory of a device to support trip planning that is partially implemented in a new release of the iPhone. As GIS presses onward, it will continue to be embedded within more and more electronics. While appliances such as refrigerators and stoves do not generally need location information, most devices that move today already have some sort of location-finding mechanism inside them. The future of these devices may rely on the ability to more precisely locate themselves, particularly inside buildings. Reuter and Zipf suggest this may come in a ground-based GPS system they term a "Global Universal Computer."

The amount of spatial data is blossoming and will likely continue to do so. As users mark important personal events and places linked to particular places, storage and retrieval of this data will become increasingly important. Reuter and Zipf suggest it is the storage and search of these items that will be most important. This technology may be crucial to historians and psychologists working to understand the reasons for individual behavior.

At some point, the lack of widespread spatial education will segment the population further, based on those who can use new integrated devices and those who cannot. Technological advances will make up for some of this digital divide by simplifying interfaces. However, these new interfaces generally cannot wholly account for such differences and maintain full functionality without a significant paradigm shift.

Finally, it may be noted that GIS in the future will become more involved with the third and fourth dimensions. The third dimension is already being implemented in geographic exploration systems and the integration of products such as Google SketchUp (http://sketchup.google.com/) into Google Earth. The fourth dimension is time, a difficult concept to incorporate into traditional GIS software structures. Peuquet[33] has provided part of the theoretical paradigm for this new implementation and new versions of commercial software such as ESRI's ArcGIS 9.3 make it easier to create animated visualizations of geodatabases. 3-D/4-D GIS will be the new frontier.

BIBLIOGRAPHY AND ADDITIONAL RELATED RESOURCES

GIS Day

On November 14, 2007, GIS Day was held in over 80 countries around the World and in all 50 states in the United States. GIS Day is a grassroots movement in which GIS users and vendors (academics,

© 2012 Taylor & Francis Group, LLC

government employees and entrepreneurs) open their doors to schoolchildren and all members of the general public in order to showcase the capabilities of GIS projects which they have developed (http://www.gisday.com/). The event is sponsored principally by the National Geographic Society, the Association of American Geographers, the University Consortium for Geographic Information Science, the United States Geological Survey, The Library of Congress, Sun Microsystems and Hewlett-Packard and ESRI, and by local GIS organizations. The next GIS Day will be held on November 18 and 17 in 2009 and 2010, respectively. The event is usually held as part of Geography Awareness Week, which has been sponsored by the National Geographic Society since 1987. The U.S. e-Government Web site using data from Daratech estimates that there are 1,000,000 users of GIS worldwide, half of whom are in the United States (http://www.whitehouse.gov/omb/egov/c-7-10-b-geospatial.html).

Books

The most important reference works for GIS are the so-called "Big-Books" of GIS. The first edition of this huge, two-volume review of the state of the art in GIS was edited by Maguire, Goodchild, and Rhind[34] and published in 1991, while the second edition was edited by Longley, Goodchild, Maguire, and Rhind[35] in 1999. More recently, the second volume has been published in a paperback edition with editorial updates based on input from the individual chapter authors, various additional chapters, and a CD featuring all the chapters from the second edition (Longley et al.[36]). Popular textbooks discussing the concepts behind GIS include Longley et al.[37] and Clarke.[4] The latter author provides an extensive list of GIS books magazines and journals, conference proceedings, and professional organizations. Price's[38] text is a guide to operating the industry leading ArcGIS 9.2 software and includes a series of hands-on tutorials to aid the novice user.

A searchable GIS bibliography may be found at ESRI's Web site: http://training.esri.com/campus/library/index.cfm. Important publishers of GIS texts include ESRI (http://store.esri.com/esri/category.cfm?SID=2&Category_ID=35) and Taylor & Francis (http://gis.tandf.co.uk/). Longley et al.[37] provide a list of major GIS textbooks while Chrisman[6] describes the earliest days of the discipline. Vendor publications have been discussed above. Suffice it to note that most major vendors have a company publication designed to inform their user base of the latest developments in their software products and many of these are now available online.

Journals and Magazines

Some of the main academic journals in which GIS research is published include

Annals of the Association of American Geographers (http://www.aag.org/)
Canadian Geographer (http://www.blackwellpublishing.com/CG)
Cartographica (http://www.utpjournals.com/carto/carto.html)
Cartographic Perspectives (http://www.nacis.org/index.cfm?x=5)
Cartography and GIS (http://www.cartogis.org/)
Computers, Environment, and Urban Systems (http://www.elsevier.com/locate/compenvurbsys)
Computers and Geosciences (http://www.elsevier.com/locate/cageo)
Conference Papers in GIS (http://srmwww.gov.bc.ca/gis/papers/index.html)
ESRI User Conference Proceedings (http://gis.esri.com/library/userconf/index.html)
Geocarto International (http://www.geocarto.com/geocarto.html)
Geographical Systems (http://link.springer.de/link/service/journals//10109/)
GeoInformatica (http://www.wkap.nl/journalhome.htm/1384-6175)
Geoscience E-Journals (http://paleopolis.rediris.es/geosciences/)
Geographical Journal (http://www.ingentaconnect.com/content/bpl/geoj/latest)
GeoJournal (http://www.ingentaconnect.com/content/klu/gejo/latest)
GIS Law

© 2012 Taylor & Francis Group, LLC

IEEE Transactions on Computer Graphics and Applications (http://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=38)
IEEE Transactions on Geoscience and Remote Sensing (http://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=36)
International Journal of Geographical Information Science (http://www.tandf.co.uk/journals/titles/13658816. asp)
International Journal of Remote Sensing (http://www.tandf.co.uk/journals/frameloader.html?http://www.tandf.co.uk/journals/tf/01431161.html)
International Journal of Mapping Sciences and Remote Sensing (http://www.ingentaconnect.com/content/bell/msrs/latest)
Journal of Geographical Systems (http://link.springer.de/link/service/journals/10109/index.htm)
Photogrammetric Engineering and Remote Sensing (http://www.asprs.org/publications/pers/www.asprs.org/publications/pers/)
Public Health GIS News and Information (http://www.cdc.gov/nchs/about/otheract/gis/gis_publichealthinfo.htm)
Remote Sensing Reviews (http://www.tandf.co.uk/journals/online/0275–7257.asp)
Transactions in GIS (http://www.blackwellpublishing.com/journals/tgis/)
The Spatial Odyssey Website also has a list of GIS Journal Abstracts and Citations (http://libraries.maine.edu/Spatial/gisweb/journals/journals.html)

Many **magazines** are available in both an online and a paper version. Some of the more notable examples are

ArcNews Online (http://www.esri.com/news/arcnews/arcnews.html)
ArcUser Online (http://www.esri.com/news/arcuser/index.html)
Challenges: A news letter from UCGIS (http://dusk2.geo.orst.edu/ucgis/news/)
Asian surveying and mapping (http://www.asmmag.com/)
GEOWorld (http://www.geoplace.com/)

Other online GIS-oriented magazines include

Directions Magazine (http://www.directionsmag.com/)
Earth Observing Magazine (http://www.eomonline.com/)
GeoCommunity (http://www.geocomm.com/)
Geomatics Information and Trading Centre (http://www.gitc.nl/)
GeoSpatial Solutions (http://www.geospatial-online.com/)
GeoVision (http://www.gisvisionmag.com/)
Geomatics Info Magazine International (http://www.reedbusiness-geo.nl/Home.asp)
GPS World (http://www.gpsworld.com/)
Spatial News (http://spatialnews.geocomm.com/)
Mentor Software (http://www.mentorsoftwareinc.com/cc/ccdir.htm)
Position Magazine (http://www.positionmag.com.au/)
Professional Surveyor Magazine Online (http://www.profsurv.com/)
The CADD/GIS Technology Center CADD/GIS Bulletins Page (https://tsc.wes.army.mil/news/bulletins/)

Organizations

The following are some of the better known organizations with a strong interest in GIS:

The American Congress on Surveying and Mapping (ACSM) (http://www.acsm.net/)
The American Society for Photogrammetry and Remote Sensing (ASPRS) (http://www.asprs.org/)
The Association for Geographic Information (AGI) (http://www.agi.org.uk/)
The Association of American Geographers (AAG) (http://www.aag.org/ this organization has a specialty group devoted to GIS) (http://geography.sdsu.edu/aaggis/)
The International Geographical Union which has a Commission on Geographical Information Science (http://www.hku.hk/cupem/igugisc/)

© 2012 Taylor & Francis Group, LLC

The North American Cartographic Information Society (NACIS) (http://www.nacis.org/)
Geospatial Information and technology Association (http://www.gita.org/)
The Urban and Regional Information Systems Association (URISA) (http://www.urisa.org/)

Conferences

This section lists a number of the more important conferences other that the vendor-specific conferences mentioned above. Many of the general, omnibus GIS Conferences have in recent years folded as more specialized offerings take their place. These conferences have usually produced either a print proceedings or a proceedings on CD-ROM.

Most of the major GIS organizations such as URISA will also have annual and even regional GIS conferences. Some conferences are strictly devoted to a single theme and are strongly oriented toward training. This is true of the Web mapping conferences (http://www.gisconferences.com/). In 2007, the following was a selection of the conferences held across the globe:

ACM GIS conference in Bellevue, Washington
Africa GIS conference in Ouagadougou, Burkina Fasa
AGIC (Arizona Geographic Information Council); GIS conference in Prescott, Arizona
AGILE (Association Geographic Information Laboratories Europe); conference on GIS in Aalborg, Denmark
Annual CA Geographic Information Association conference in Cypress, California
Annual GIS conference, ASPRS and URISA in Vancouver, Washington
Annual GIS for Oil & Gas Conference in Aurora, Colorado
Annual International airport GIS conference in Budapest, Hungary
Annual Minnesota GIS conference in Rochester, Minnesota
Annual Missouri GIS conference in Osage Beach, Missouri
Annual NC GIS conference in Winston-Salem, North Carolina
Annual Ohio GIS conference in Columbus, Ohio
Annual Virginia GIS conference in Virginia Beach, Virginia
Arc GIS conference in Biloxi, Mississippi
Biennial GIS conference, Iowa Geographic Council in Sioux City, Iowa
California GIS conference in Oakland, California
Croatian GIS Association conference in Sinj, Croatia
Delaware GIS conference in Dover, Delaware
East Tennessee GIS conference in Pigeon Forge, Tennessee
Eastern Montana GIS conference in Miles City, Missouri
ESRI Asia-Pacific User Conference in New Delhi, India
ESRI Australia: GIS user conference in Sydney, Australia
ESRI Eastern Africa: GIS user conference in Kampala, Uganda
ESRI Federal Users GIS conference in Washington, District of Columbia
ESRI GIS solution expo in Danvers, Massachusetts
ESRI Health GIS conference in Scottsdale, Arizona
ESRI International User conference in San Diego, California
ESRI New Zealand: GIS user conference in New Zealand
ESRI South Asia user conference in Novotel Clarke Quay, Singapore
EUC (European User Conference) in Stockholm, Sweden
The GeoTec Event in Ottawa, Ontario, Canada
GI and GIS conference in Porto, Portugal
GIS conference, Office of Lt Governor, U.S. Virgin Islands
GIS Engineers Society conference in Trivandrum, India
GIS for public sector conference in London, U.K.
GIS for Urban Environmental summit in Johannesburg, South Africa
GIS in Rockies conference in Denver, Colorado
GIS in Transit in Tampa, Florida
GIS South Africa conference in Umhlanga Rocks, Durban
Historical GIS conference in Nagoya, Japan

© 2012 Taylor & Francis Group, LLC

Homeland Security GIS summit in Denver, Colorado
Illinois GIS conference (ILGISA) in Oak Brook, Illinois
Indiana GIS conference in Indianapolis, Indiana
Indonesian Geospatial Technology Exhibition in Jakarta, Indonesia
Intermountain GIS conference in Donnelly, Idaho
International conference of GIS/RS in Hydrology in Guangzhou, China
International conference on Health GIS in Bangkok, Thailand
International GIS crime mapping conference in Brussels, Belgium
Ireland GIS conference in Dublin, Ireland
Kentucky GIS conference in Louville, Kentucky
Kuwait GIS conference in Kuwait
Map Asia in Kulamanpur, Malaysia
Memphis Area Geographic Information Council GIS conference in Memphis, Tennessee
National GIS symposium in Saudi Arabia in Khobar, Saudi Arabia
Nebraska GIS Symposium in Omaha, Nebraska
Nordic GIS conference in Herning, Denmark
North Dakota GIS user conference in Bismarck, North Dakota
North Western PA GIS conference in Clarion, Pennsylvania
Northeast Arc Users Group: GIS conference in Burlington, Vermont
NSGIC (National States Geographic Information Council); in Madison, Wisconsin
NYS GIS conference in Liverpool, New York
PA GIS conference in Harrisburg, Pennsylvania
Pacific Islands GIS/RS conference in Suva, Fiji
Real estate GIS user conference in Scottsdale, Arizona
Rhode Island GIS conference in Narragansett, Rhode Island
ScanGIS—Scandinavian GIS Conference in As, Norway
Southern Forestry and Natural Resources Management GIS conferences in Kissimmee, Florida
Super map GIS conference in Beijing, China
Towson GIS conference in Towson, Maryland
UGIC (Utah Geographic Information Council)—GIS conference in Salt Lake City, Utah
URISA & IAAO 11th Annual GIS conference in Las Vegas, Nevada
URISA (urban regional information systems association)
VIGIC (Virgin Islands Geospatial Information Council)
Washington GIS conference in Lynnwood, Washington

GIS Dictionaries

The Association for Geographic Information has an online dictionary at http://www.geo.ed.ac.uk/agidict/welcome.html. For a published GIS dictionary McDonnell and Kemp's[39] International GIS Dictionary can be referred to.

ACKNOWLEDGMENT

The authors would like to thank Matt Ball for comments on an earlier draft.

REFERENCES

1. Waters, N.M. Geographic information systems. In *Encyclopedia of Library and Information Science*; Marcel Dekker Inc.: New York, 1998; Vol. 63, Supplement 26, 98–125.
2. Waters, N.M. Geographic information systems. In *Encyclopedia of Library and Information Science*, 2nd Ed.; Drake, M., Ed.; Marcel Dekker, Inc.: New York, 2003; 1106–1115.
3. Dibiase, D.; Demers, M.; Johnson, A.; Kamp, K.; Taylor Luck, A.; Plewe, B.; Wentz, E. *Geographic Information Science and Technology Body of Knowledge*; Association of American Geographers: Washington, DC, 2006.
4. Clarke, K.C. *Getting Started with Geographic Information Systems*, 4th Ed.; Prentice Hall: Upper Saddle River, NJ, 2003.

© 2012 Taylor & Francis Group, LLC

5. Foresman, T.W., Ed.; *The History of Geographic Information Systems: Perspectives from the Pioneers*; Prentice Hall: Upper Saddle River, NJ, 1997.
6. Chrisman, N. *Charting the Unknown: How Computer Mapping at Harvard Became GIS*; ESRI Press, Redlands, CA, 2006.
7. Burrough, P.; McDonnell, R. *Principles of Geographical Information Systems*, 2nd Ed.; Oxford University Press: New York, 1998.
8. Longley, P.; Goodchild, M.F.; Maguire, D.J.; Rhind, D.W. Introduction. In *Geographical Information Systems, Vol. 1, Principles and Technical Issues*; Longley, P., Goodchild, M.F., Maguire, D.J., Rhind, D.W., Eds.; Wiley: New York, 1999; 1–16.
9. Kang, H.; Scott, D.M. An integrated spatio-temporal GIS toolkit for exploring intra-household interactions. *Transportation*, **2008**, *35*, 253–268.
10. Mark, D.M. Geographic information science: Defining the field. In *Foundations of Geographic Information Science*; Duckham, M., Goodchild, M.F., Worboys, M.F., Eds.; Taylor & Francis: New York, 2003; 3–18.
11. Wright, D.J.; Goodchild, M.F.; Proctor, J.D. Demystifying the persistent ambiguity of GIS as 'tool' versus 'science.' Ann. Assoc. Am. Geogr. **1997**, *87*, 346–362.
12. Chrisman, N.R. What does GIS mean? Trans. GIS **1999**, *3*, 175–186.
13. Taylor, D.R.F., Ed.; *Policy Issues in Modern Cartography*; Elsevier Science: Oxford, 1998.
14. Craig, W.J.; Harris, T.M.; Weiner, D. *Community Participation and Geographical Information Systems*; CRC Press: Boca Raton, FL, 2002.
15. Goodchild, M.F. Citizens as sensors: The world of volunteered geography. GeoJournal **2007**, *69*, 211–221.
16. National Research Council, *Learning to Think Spatially*; The National Academies Press: Washington, DC, 2006.
17. Bednarz, S.W.; Bednarz, R.S. Geography education: The glass is half full and it's getting fuller. Prof. Geogr. **2004**, *56*, 22–27.
18. Doering, A.; Veletsianos, G. An investigation of the use of real-time, authentic geospatial data in the K-12 classroom. J. Geogr. **2007**, *106*, 217–225.
19. McClurg, P.A.; Buss, A. Professional development: Teachers use of GIS to enhance student learning. J. Geogr. **2007**, *106*, 79–87.
20. Marsh, M.; Golledge, R.; Battersby, S.E. Geospatial concept understanding and recognition in G6-college students: A preliminary argument for minimal GIS. Ann. Assoc. Am. Geogr. **2007**, *97*, 696–712.
21. Kwan, M.P.; Knigge, L. Guest editorial: Doing qualitative research using GIS: An oxymoronic endeavor? Environ. Plann. A **2006**, *38*, 1999–2002.
22. Matthews, H. Inaugural editorial: Coming of age for children's geographies. Child. Geogr. **2003**, *1*, 3–5.
23. Dennis, S. Prospects for qualitative GIS at the intersection of youth development and participatory urban planning. Environ. Plann. A **2006**, *38*, 2039–2002.
24. Noble, D. *Digital Diploma Mills*; Monthly Review Press: New York, 2003.
25. Limp, W.F. ArcGIS extensions: Quick take review. Geo-World **2005**, *18* (7), 54–58.
26. Saaty, T.L. *Theory and Applications of the Analytic Network Process: Decision Making with Benefits, Opportunities, Costs, and Risks*; RWS Publishers: Artarmon, Australia, 2005.
27. Ball, M. Digital reality: Comparing geographic exploration systems, 2006. http://www.geoplace.com.
28. Gould, P.R. Is Statistix Inferens the geographical name for a wild goose? Econ. Geogr. **1970**, *46*, 439–448.
29. Anselin, L. *Spatial Econometrics*; Kluwer: Dordrecht, 1988.
30. Anselin, L. Local indicators of spatial autocorrelation. Geogr. Anal. **1995**, *27*, 93–115.
31. Anselin, L.; Florax, R. *New Directions in Spatial Econometrics*; Springer-Verlag: Berlin, 1995.
32. Fotheringham, A.S.; Brunsdon, C.; Charlton, M. *Quantitative Geography: Perspectives on Spatial Analysis*; Sage: London, 2000.
33. Peuquet, D. *Representations of Space and Time*; Guilford: New York, 2002.
34. Maguire, D.J.; Goodchild, M.F.; Rhind, D.W. Eds., *Geographical Information Systems*; Longman: London, 1991.
35. Longley, P.; Goodchild, M.F.; Maguire, D.J.; Rhind, D.W. Eds., *Geographical Information Systems*; Wiley: New York, 1999.
36. Longley, P.A.; Goodchild, M.F.; Maguire, D.J.; Rhind, D.W. Eds., *Geographical Information Systems*, 2nd abridged Ed.; Wiley: New York, 2005.
37. Longley, P.A.; Goodchild, M.F.; Maguire, D.J.; Rhind, D.W. *Geographic Information Systems and Science*, 2nd Ed.; Wiley: New York, 2005.
38. Price, M. *Mastering ArcGIS 9.2*; McGraw-Hill: New York, 2008.
39. McDonnell, R.; Kemp, K. *International GIS Dictionary*; Longman: London, 1995.

© 2012 Taylor & Francis Group, LLC

36 Clinical Decision-Support Systems

Kai Zheng

CONTENTS

INTRODUCTION

The notion that artificial intelligence (AI) might one day rival the decision-making capability of human brain sparked the first generation of computerized clinical decision-support systems (CDSS) developed from the 1960s into the 1980s.[1,2] However, years of trials and frustrations convinced AI enthusiasts that the enormous variations in patient care cannot be reduced to systematic decision making to render qualitative medical treatments.[3] These limitations became even more apparent with the increasing awareness that patient care outcomes are also subject to other factors including quality- and value-of-life judgments, economic and psychosocial considerations, and social well-being of the patient as a whole. This brought a disappointing close to the first chapter in the use of computers to aid in medical decision-making.[4]

The story did not end there. Computerized clinical information systems proliferate throughout health care organizations today, significantly reducing the costs to acquire and store patient data. These changes, however, have invited new challenges, including an explosion of patient information that far exceeds any practitioner's capability of processing such data.[5] This situation is further compounded by a wellspring of medical knowledge resulting from the "evidence-based medicine" (EBM) movement over the past 20 years that has revolutionized the way medicine is practiced. EBM requires physicians to rely less on their own experience and more on the current best evidence in making decisions about the care of individual patients.[6] Unfortunately, "current best evidence" is a temporal concept that may become outdated rapidly as medical research advances or the mechanisms causing diseases change (e.g., new varieties of flu virus).

Further, the U.S. health care system has been criticized for its high costs, low efficiency, poor quality, and unacceptable rates of preventable medical errors. In 2007, $2.2 trillion or 16.3% of national gross domestic product (GDP) was spent on health care in the United States,[7] while this

© 2012 Taylor & Francis Group, LLC