In *Proceedings of the Eighth International Conference on Information Knowledge Management, (CIKM-99), pp. 210-216,*

Architecture of a Metasearch Engine that Supports User Information Needs

Eric J. Glover[1,2], Steve Lawrence[1], William P. Birmingham[2], C. Lee Giles[1]

{compuman,lawrence,giles}@research.nj.nec.com[1]
{compuman,wpb}@eecs.umich.edu[2]

NEC Research Institute[1]
4 Independence Way
Princeton, NJ 08540

Artificial Intelligence Laboratory[2]
University of Michigan
1101 Beal Avenue
Ann Arbor, MI 48109

Abstract

When a query is submitted to a metasearch engine, decisions are made with respect to the underlying search engines to be used, what modifications will be made to the query, and how to score the results. These decisions are typically made by considering only the user's keyword query, neglecting the larger information need. Users with specific needs, such as "research papers" or "homepages," are not able to express these needs in a way that affects the decisions made by the metasearch engine. In this paper, we describe a metasearch engine architecture that considers the user's information need for each decision. Users with different needs, but the same keyword query, may search different sub-search engines, have different modifications made to their query, and have results ordered differently. Our architecture combines several powerful approaches together in a single general purpose metasearch engine.

1 Introduction

Current metasearch engines make several decisions on behalf of the user, but do not consider the user's complete information need when making these decisions. A metasearch engine must decide which sources to query, how to modify the submitted query to best utilize the underlying search engines, and how to order the results. Some metasearch engines allow users to influence one of these decisions, but not all three.

The primary advantages of a metasearch engine over a single search engine are increased coverage and a consistent interface [16]. A recent study by Lawrence and Giles [12] estimated the size of the web at about 800 million indexable pages. This same study concluded that no single search engine covered more than about sixteen percent of the total. By searching multiple search engines simultaneously via a metasearch engine, coverage increases dramatically over searching only one engine. Lawrence and Giles found that combining the results of 11 major search engines increased the coverage to about 42% of the estimated size of the publicly indexable web.

A consistent interface is necessary for a metasearch engine to be useful [4, 10]. Such an interface ensures that results from several places can be meaningfully combined, while insulating the user from the specifics of the underlying search engines.

In this paper, we describe the architecture of the next generation of Inquirus, the metasearch tool at NEC Research Institute. This architecture, shown in Figure 3, makes certain user preferences explicit. These preferences define a search strategy that specifies source selection, query modification, and result scoring. Allowing the user to control the search strategy can provide relevant results for several specific needs, with a single consistent interface.

A typical metasearch engine (Figure 2), such as DogPile [1], submits a user's query (with minor modifications for syntactic consistency) to a set of search engines, and returns the results in the order returned by the search engines. This order might not make sense if the user has a specific need, such as "current events" or "research papers."

User's information needs are not sufficiently represented by a keyword query alone. Studies have shown that users consider many factors, including some which are non-topical, when making relevance judgments [3, 15].

Glover and Birmingham [7] demonstrated the use of decision theory as a means of re-ordering results from a single

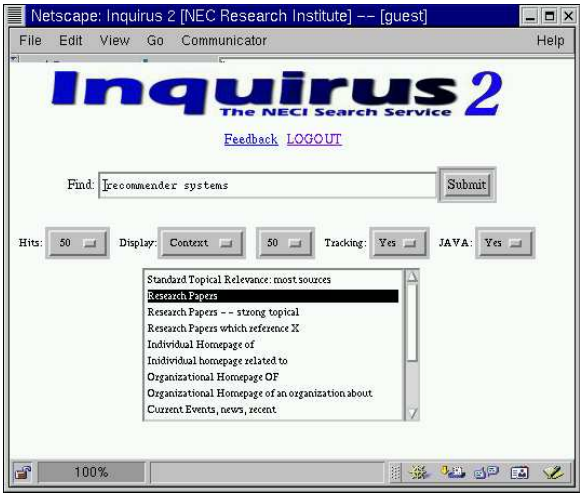

Figure 1: Screen shot of the Inquirus 2 interface

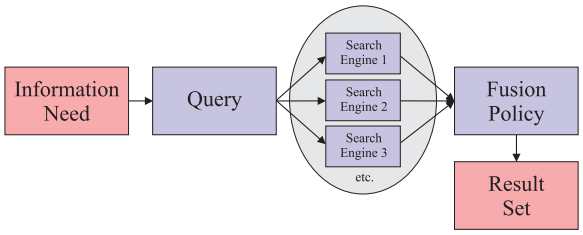

Figure 2: The architecture of a standard metasearch engine

search engine while capturing more of a user's information need than a text query alone. Nguyen and Haddawy [14] have also demonstrated the use of decision theory as a means of making result ordering decisions.

In addition to properly ordering the results, choosing where to look affects the overall search precision. To increase the precision of the results, some metasearch engines such as ProFusion [5], SavvySearch [8] or MetaSEEK [4] do not always send the user's query to the same search engines. ProFusion considers the performance, the predicted subject of the query and the user's explicit search engine preferences. On request, ProFusion downloads individual pages to check for broken links and duplicates. MetaSEEK considers past results and the user's keywords for source selection, and the current version of SavvySearch allows users to specify a "category" to determine the sources searched.

Metasearch engines can significantly increase coverage, but are limited by the engines they use with respect to the number and quality of results. It is important that queries sent to the underlying search engines best reflect the current need of the user. Many popular Internet search engines, such as HotBot [2], allow users to add extra (non-topical) constraints, such as "in the last week" or "language must be English." Likewise, users of a regular (or metasearch) engine might add extra (non-topical) terms in the hopes of increasing the precision, such as adding the term "home" to a query looking for someone's homepage. Inquirus [11], the metasearch engine of the NEC Research Institute, will automatically recommend simple query modifications, such as the use of phrases or requiring optional terms if there are too many results. Inquirus will also take simple phrases, such as "What is X," and modify it to better reflect the intended meaning.[1] It is desirable for a metasearch engine to perform dynamic query modifications to maximize the result quality from a search engine.

Query modifications can increase coverage. As an example, a user would not normally consider HotBot when looking for news, since it is a general-purpose search engine. However, with a date constraint, it is possible to get many valuable (both topically relevant and recent) results, not available from a news-specific search engine. On the other hand, a user searching for "research papers" would likely rule out good results by adding a constraint on date. The use of such options depends on the user's need, and is applied differently to different search engines. Section 2.2 describes the query modifications used by Inquirus 2 in more detail.

2 A new architecture

Figures 1 and 3 show the interface and architecture of the next generation of Inquirus. This architecture has an explicit notion of user preferences, which includes preferences over non-topical features. These preferences, or a search strategy, are used to choose the appropriate search engines and query modifications, and influence the order the results. The current user interface to Inquirus 2, Figure 1, provides the user with a list of choices (every user could have their own customized list). Such choices currently include:[2] research papers, general introductory, individual homepage of, organizational homepage of, etc.

The specification of preferences allows users with different needs, but the same query, to not only search different search engines (or the same search engines with different "modified" queries), but also have results ordered differently. An example is one user searching for research papers about "information retrieval," versus a second user looking for homepages of organizations specializing in "information retrieval." Even though both users have different information needs, they might type the same keyword query, and even search some of the same search engines. Tables 3 and 4 show actual results for this query, using the Inquirus 2 system, for the different information needs.

Our new architecture was built on top of Inquirus [11], which guarantees consistent scoring of results by download-

[1] Inquirus automatically changes the query "What is X" to "X is" "X refers to" "X means" "X will" "X helps".

[2] As of this writing, all the search strategies were human generated, however our future work plans on using learning to improve them.

Name	Description	Search engines used
Research papers	Detailed pages, preferably an actual article	Google* AltaVista Snap Yahoo* HotBot NorthernLight
Individual homepages	The homepage(s) of the individual listed in the query	Snap Google HotBot Yahoo
Organizational homepage of	The homepage(s) of the organization listed in the query	Snap Google HotBot Yahoo
Organizational homepage about	Homepages of organizations related to the query, i.e. doing research on, selling the product, etc.	Snap Google HotBot Yahoo
Current events, news recent	Recent articles, or content about the given query, with significant content	ABCNews News.com Snap AltaVista Yahoo HotBot*
Current events, more topical	Same as above, but stronger weight on topical relevance	ABCNews News.com Snap AltaVista Yahoo HotBot*
General introductory about	Getting started, references, "What is", etc.	Google* AltaVista* Snap Yahoo

Table 1: Information need categories and the search engines used. * means query sent to the search engine is modified to enhance precision

Name	Description
agrade	Average of three grade level algorithms, FOG, SMOG, and FK
GFOG	A reading level algorithm optimized for less advanced documents
daysOld	The predicted number of days old as computed by analyzing the full text and HTML (not only considering the header)
wordcount	The number of words per page
homepage	A measure of the number of homepage like features present
genscore	A measure of features indicative of a "general" page, such as the keywords "links" or "resources"
researchpaper	A measure of features indicative of a "research paper" page, such as having an abstract or references
anchorcount	The number of unique links present on a page
imagecount	The number of unique images present on a page
numkeywords	The number of keywords in the query matched on a page
sectioncount	The number of sections on a page
pathlength	The depth of a page from the top of a domain in levels
summary	An automatically generated summarization of the document
topicalrelevance	A query dependent attribute predicting how much a particular page is "about" the given query. The attribute is based on word distances, from each other and the top of the document, as well as the number of occurrences of each term

Table 2: List of some of the page specific attributes and their description

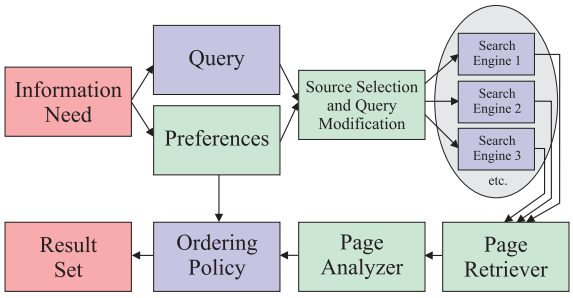

Figure 3: The architecture of the Inquirus 2 search engine

ing page contents and analyzing the pages on the server, as opposed to relying on the reported scores and short summaries from the original search engines. By downloading pages locally, we are assured to have the most recent version of any page and eliminating dead links and old content.

Inquirus 2 opens up three decisions, source selection, query modification, and document scoring. Each decision is described in detail below.

2.1 Choosing a search engine

The preferences specify the sources used, as shown in Figure 3. Currently, the search strategies are human generated. Our future work includes use of learning techniques to improve the source selection decision. Several metasearch engines, MetaSEEK, ProFusion, SavvySearch, and others, allow the user some control of which search engines are chosen. Each of these engines has had experimental methods for automatic source selection based on the user query. MetaSEEK [4] considers the performance history for given keywords, while ProFusion [5] considers the predicted subject of the query. SavvySearch [8] currently allows users to choose a "category" and uses search engines specific for the given category, but had a prototype with automatic source selection based on the predicted recall for a given query.

A difficulty of automatic source selection is the metric used to compare sources. One metric is the number of results

returned (on average) for a given query or subject.[3] This metric is similar to the recall measure used in IR, if one assumes all returned results are *relevant*. Unfortunately, there is not necessarily a correlation between number of results returned and usefulness of those results. As a simple example, a user searching for "news" about President Clinton may find fewer results on a site dedicated to news than a large general-purpose search engine, but the results from the news site are likely (although not necessarily) more valuable to the user.

MetaSEEK attempted to consider user satisfaction with results for some given query, in a sense a collaborative definition of the sources used based on user feedback. Although this approach considers user statements about result values, it assumes that the user's valuations are mapped to the query, which is not true if there is more than a one-to-one mapping of needs to queries. For example, just because all previous users felt photos of real dogs were "good", and cartoon dogs were "bad" does not mean that a new user will agree.

Part of the search strategy of Inquirus 2 contains a function for predicting the value of documents. This function is specific to the individual user. Given a reasonable model of user value, it is possible to determine how good any given source is for a given need (on average), as opposed to associating the "worth" of a given source to a query. By evaluating the worth of a source based on the need, not the query, it is possible to make reasonable judgments for previously unseen queries. A user looking for news will usually prefer results from a site dedicated to news, rather than a result from a general-purpose search engine, regardless of their query.

2.2 Modifying the query

One of the problems of a metasearch engine is its dependence on the underlying search engines to provide a reasonable set of results. Just because a search engine contains very good results for the current user's need, there is no guarantee that those results will be returned for any given query. A simple example is a user searching HotBot for "current events" about Linux. When given a query of "Linux," the search engine may return thousands of pages which, although about Linux, are not recent, or news oriented. To compound the problem, many search engines limit the number of results returned to the user, such as AltaVista's limit of 200 URLs. This limit can result in none of the "good" pages ever being seen by the user.

To enhance the precision of the results, and deal with the problems caused by search engine result limits, Inquirus 2 allows query modification. There are three types of modification performed: Use of the search engine specific options, such as sort by date, or constrain to a language; prepending terms; or appending terms. Query modification is one method of causing the underlying search engines to provide more valuable results for a given information need.

One method for modification is taking advantage of search-engine-specific options. Most search engines provide the ability to influence their result ordering, or to add constraints. One example is the addition of a constraint on language. A second example is instructing the search engine to sort results by date. The choice of options depends on the user's information need, not only his keyword query.

The second modification adds keywords. Depending on the user's information need, it might be desirable to locate pages by "type." For example, a research paper will typically contain the sections abstract, introduction, and references. By adding those three keywords to a query, the density of research papers (similar to precision) can be significantly increased. It is important to note that query modification does not affect the scoring function, documents are scored based on the user-provided query and preferences, not the modification. Inquirus 2 allows query terms to be added before or after the provided query. For the information need category of "general resources," both types of modification are used. One modification is prepending "what is" to the query, while another adds the keywords "resources links."

The effects of dynamic query modification can be seen in Tables 3 and 5, where several of the top ranked results were found through the use of modified queries.

2.3 Ordering results

Probably the most important decision made by a metasearch engine is how to order the results. A typical search engine scores results based on the keywords in the query and the terms in the document [13]. Typical metasearch engines score documents based on the original scores returned from the search engines queried, running the risk that the actual pages are no longer relevant, or that the page scored high as a result of keyword spamming[4]. Inquirus and Inquirus 2 download every web page and order them based on the full content. Inquirus 2 improves upon the ordering policy of Inquirus by using an ordering policy defined by the user's preferences, as shown in Figure 3. Different users, even with the same query and the same set of documents, will have results presented in an order meaningful to their individual need.

[3] ProFusion, and the work done in SavvySearch relied on number of results returned from any given search engine.

[4] Keyword spamming is an attempt by content providers to cause their page to be ranked highly by actively altering the HTML to take advantage of the scoring functions used by the search engines.

#	Source	Title	Comment
1	Google*	Geographic Information Retrieval and Spatial Browsing	A research paper about Geographic IR and spatial browsing
2	Google*	Information Retrieval of Imperfectly Recognized Handwriting	A research paper from 1993 about handwriting as the interface to IR
3	NLight	Content-based indexing and retrieval of video	Actual page title was different, this is the paper title. The paper is about a proposal for a video IR system
4	Google*	High-Performance, Distributed Information Retrieval	A proposal for a high-performance, distributed search engine called "KEYNET"
5	NLight	Intelligent Multimedia Information Retrieval Workshop	Not actually a research paper, but rather a call for papers
10	Yahoo*	Register of Ecological Models:GOSSYM	This document, formatted like a research paper describes a cotton growth model and expert system.

Table 3: Results for the query 'information retrieval', with a preference of 'Research Papers', * means a modified query found the result

We treat the document-ordering task as a decision problem, and use utility theory [9] for evaluating the results. The ordering policy is "sort by value," where utility theory provides the mechanism for predicting the value. Each user-selected information need category has an associated additive value function of the form shown in Equation 1.

$$\mathcal{U}(d_j) = \sum_k w_k v_k(x_{jk}) \quad (1)$$

Where w_k is the weight of the k_{th} attribute, and by convention totals one. v_k is the value function for the k_{th} attribute, x_{jk} is the level of the k_{th} attribute for the j_{th} document, and by convention: $\forall k, d : v_k(d) \in [0, 1]$.

The page analyzer extracts the attributes (x_{jk}) for every page. Table 2 lists several of our page specific attributes. Each information need category is described as a value function allowing a balance between several attributes as opposed to considering only one. Glover and Birmingham [6] demonstrated the feasibility of this approach for re-ordering web pages from HotBot. The DIVA system [14] also uses utility theory to order alternatives.

3 Dynamic interface

Inquirus 2's method of downloading all web pages can be time consuming, and presents an interface issue. If the ordering policy is sort by value, the standard approach is to wait until all results are downloaded and scored, then sort. An alternative is a dynamic interface that inserts each result as it is scored. If by coincidence the very first downloaded web page is "perfect," it will be immediately available, and the user can stop the search. Likewise, if there are mixtures of "good" and "bad" results, the "better" ones will be displayed on top. As a new result is downloaded and scored, it is immediately available for the user to see, thus reducing the effect of the latency, and improving over many metasearch engines that force the user to wait before seeing results.

Inquirus 2 provides the user with an optional JAVA applet that provides this dynamic-sorting and display functionality, plus the ability to change the ordering policy during or after searches. As results are coming in, a user can tell the applet to sort by "topical relevance" or "sort by date" as opposed to sorting by the original information need category. The new sort criterion is immediately effective, and persists as new results are found.

4 Some results

Tables 3, 4 and 5 list several results for the query "information retrieval," with three different preferences. Currently, we supplied the utility functions, but our architecture allows anyone to define new information need categories and corresponding utility functions. If a user has a different notion of what they mean by research papers, their individual functions can be personalized. The sections below describe our selection of attributes, and their value functions based on our information need category.

4.1 Research papers

A user searching for research papers cares about how relevant the paper (web page) is to their query, as well as the content and format of the page. The perfect result might be a web page that is a complete journal article, relevant to the query. The associated value function for this need has a 75% weight on topical relevance, and 25% of the weight divided among the average grade level, the *researchpaper* attribute, and the wordcount. Table 3 shows several actual results. 200 web pages were downloaded and analyzed from six different search engines. The top three results shown are all research papers about some aspect of "information retrieval." The fourth result, although not a research paper,

#	Source	Title	Comment
1	HotBot	Datagold Limited – a leading information services company...	Datagold Limited is a leading UK provider of information services including information retrieval software. This page is the company homepage
2	Google	The Glasgow IR Group	The information retrieval group led by Keith van Rijsbergen at University of Glasgow. This page is not the homepage of the University of Glasgow, but is for the IR group.
3	Google	Welcome to CNIDR	The homepage for the Center for Networked Information Discovery and Retrieval

Table 4: Top three results for the query 'information retrieval', with a preference of 'Organizational homepages about'

#	Source	Title	Comment
1	Google*	Site Search Tools – information retrieval Research	A list of major topics and research areas in IR and a list of links includes: TREC, Z39.50, Web IR and IE, and a list of academic research sites, originally ranked 25
2	Snap	Information Processing and retrieval Directory	A page containing ten links to various IR resources including: reference, search related, information services and news related to IR, originally ranked 177
3	Yahoo	XML.com – information search and retrieval	"Information Search and Retrieval-oriented parsers, white papers, specifications, and implementations"
7	Yahoo, Snap	Yahoo!Reference ... Information Retrieval	The Yahoo category on Information Retrieval
8	Yahoo, Snap	Information Retrieval and Digital Libraries	A collection of links about IR and Digital Libraries
9	Snap	Information Retrieval	A collection of links related to specific IR research: Boolean searching and Precision vs. Recall

Table 5: Results for the query 'information retrieval', with a preference of 'General introductory about', * means a modified query found the result

is a proposal for a search system, which is much like a research paper.

Results from four different search engines scored in the top ten for this query. Of the top ten results, four were found as a result of modified queries. The queries sent to Google and Yahoo were modified by adding "abstract keywords references" to the provided user query. Without query modification, it is unlikely that the top two results would have been found in this case.[5]

The importance of considering more than one attribute can be seen by examining the tenth-ranked result. This document, although more like a research paper than the fifth-ranked document, was less "about" information retrieval. The weights and attribute-value functions can be easily adjusted to change the importance of any given attribute, thus switching the order of these two documents. In the future, we will use learning to determine the "best" weights and attribute-value functions.

4.2 Organizational homepages about

A second information need category we describe is "organizational homepages about." The "perfect" page for this category is the homepage of an organization strongly related to the query terms. To capture this need, the preference places a 45% weight on the *homepage* attribute, and a 30% weight on the *pathlength* attribute, with the remaining 25% on *topical relevance*, query terms in the meta-keyword tags, and terms appearing in the title.

The results shown in Table 4 demonstrate how important it is to consider multiple factors as preferences as opposed to constraints. The second and third-ranked pages did not contain the words "home" or "homepage" in their title, and the second ranked page, "Glaskow IR Group" was not a top-level page. The pages scored highly because they were the best overall, even though they were not "perfect" with respect to all the attributes considered.

We do not currently consider "popularity" when ranking results. If we did have such an attribute, it could be easily incorporated to push more "popular" pages higher up. The results presented in Table 4 resulted from retrieval of 500 total web pages with none of the queries submitted modified.

4.3 General introductory

The final category is "General introductory about." For this category, a user may be searching for a starting point in some topic or trying to learn about some concept. Table 5 shows several results for this query. 500 total documents

[5] Although these same results may, in theory, be found by the same search engine with an unmodified query, they might not be accessible as a result of limits on the number of results returned. Even if listed, they would be unlikely to have been retrieved, since for this query we only retrieved 200 total results.

were retrieved from four different search engines. Of the top-ten ranked results, all but one search engine, AltaVista, was represented, and the first-ranked result was from a modified query. To find more "general" documents, we used two different query modifications. One modification added the terms "links resources" to the end of the user provided query. The other prepended the user query with the quoted string "what is." The addition of "links resources" is intended to find pages that are good starting points, i.e., pages of links and resources about the query. Adding "what is" is intended to find pages that address the question of "what is X" where X is the query.

For this information need category, we used a 20% weight on the attributes *genscore* and *topical relevance*, a 15% weight on the attributes *GFOG* and *wordcount*, and the remaining 30% on various keyword specific attributes, such as the percent of the keywords in the meta tag "keywords."

5 Summary and future work

We have described a new metasearch engine architecture in use at NEC Research Institute. This architecture utilizes user preference information in deciding where to search, how to modify the queries, and how to order the results. This approach allows for much greater personalization and higher quality results than a regular metasearch engine, because of the ability to consider more than just the keyword query when making search decisions.

References

[1] DogPile metasearch engine. http://www.dogpile.com/.

[2] HotBot search engine. http://www.hotbot.com/.

[3] Carol L. Barry. *The Identification of User Criteria of Relevance and Document Characteristics: Beyond the Topical Approach to Information Retrieval*. PhD thesis, Syracuse, 1993.

[4] Ana B. Benitez, Mandis Beigi, and Shih-Fu Chang. Using relevance feedback in content-based image metasearch. *IEEE Internet Computing*, 2(4):58–69, 1998.

[5] Susan Gauch, Guihun Wang, and Mario Gomez. ProFusion: Intelligent fusion from multiple, distributed search engines. *Journal of Universal Computer Science*, 2(9), 1996.

[6] Eric J. Glover and William P. Birmingham. Using decision theory to order documents. In *Digital Libraries 98*, Pittsburgh, PA, 1998. ACM Press.

[7] Eric J. Glover, William P. Birmingham, and Michael D. Gordon. Improving web search using utility theory. In *Web Information and Data Management (WIDM'98)*, pages 5–8, Bethesda, MD, 1998. ACM Press.

[8] Adele E. Howe and Daniel Dreilinger. SavvySearch: A meta-search engine that learns which search engines to query. *AI Magazine*, 18(2), 1997.

[9] Ralph L. Keeney and Howard Raiffa. *Decisions with Multiple Objectives*. John Wiley and Sons, New York, 1976.

[10] Steve Lawrence and C. Lee Giles. Context and page analysis for improved web search. *IEEE Internet Computing, July-August*, pages 38–46, 1998.

[11] Steve Lawrence and C. Lee Giles. Inquirus, The NECI Meta Search Engine. In *WWW7*, pages 95–105, Brisbane, Australia, 1998.

[12] Steve Lawrence and C. Lee Giles. Accessibility of information on the web. *Nature*, 400(July 8):107–109, 1999.

[13] Michael L. Mauldin. Lycos: Design choices in an Internet search service. *IEEE Expert*, (January–February):8–11, 1997.

[14] Hien Nguyen and Peter Haddawy. The Decision-Theoretic Video Advisor. In *AAAI Workshop on Recommender Systems*, 1998.

[15] Linda Schamber, Michael B Eisenberg, and Michael S. Nilan. A re-examination of relevance: Toward a dynamic, situational definition. *Information Processing and Management*, 26(6):755–776, 1990.

[16] E. Selberg and O. Etzioni. The MetaCrawler architecture for resource aggregation on the Web. *IEEE Expert*, (January–February):11–14, 1997.