TECHNOLOGY TRENDS

A New Approach To Search

Joe Weinman

Search algorithms today are largely based on a common paradigm: link analysis. But they've ignored a mother lode of data: The network.

There is a joke about an inebriated individual searching late one night for his car keys under a street lamp. When asked whether that is where he dropped them, he replies, "No. I lost them in those bushes over there, but the light's better over here." Virtually all search engines today rely on hyperlink—also known as "citation"—analysis to maximize results relevancy. But is that the best approach? Or do they do it just because, like the area under the street lamp, "the light's better" by the hyperlinks?

This article describes an innovative architecture and ecosystem for search that may offer substantial advantages over prior approaches. Rather than treating the network as a passive conduit for search engines to interact with Web servers and for browsers to interact with query processors, this approach leverages aggregate network data, metadata and statistics to better rank search results according to relevancy and enhance the scope of search to include the "Deep Web," in the process potentially reducing click fraud.

While there are probably hundreds, if not thousands of variations of search algorithms, architectures and tweaks [see endnotes 1, 2], Web search has been based largely on three canonical approaches. A first type of approach is human-powered directories, such as the original service provided by Yahoo!, now called Yahoo! Directory, where Web resources are hierarchically organized into taxonomies, *e.g.*, Leisure and Entertainment -> Travel -> U.S. -> Hawaii -> Kauai -> Beaches. A variation of this approach, used by the Open Directory Project (dmoz.org), enlists volunteers, rather than employees, to organize resources.

A second approach is crawler-based index generation, which has evolved to generate results largely prioritized by link analysis. Two popular sites that exemplify this approach are Google.com and Ask.com, which are based on the PageRank and HITS citation analysis algorithms, respectively (described later).

A third approach is collaborative tagging, such as that used by Del.icio.us, where surfers tag pages with keywords such as "winery" and "Napa" so future searchers can find those pages by entering those tags. Some engines combine variations of these approaches: Technorati.com leverages tags entered by bloggers and Web page designers, but also provides citation count ranking information (which they call "Authority").

A new approach, described in detail herein, is to use data and metadata associated with network transport of Web content—including HTML pages, documents, spreadsheets and so forth—to replace or augment traditional Web crawlers, improve the relevance and currency of search results ranking, and reduce click fraud. In the Web ecosystem, this would be of value to Web surfers, who want up-to-the-minute, useful results returned; search engine companies, wanting to maximize the value of results to their users; websites that want their information to be appropriately found by users with interest in and intent to use that information; and advertisers, who want to pay only for valid clicks.

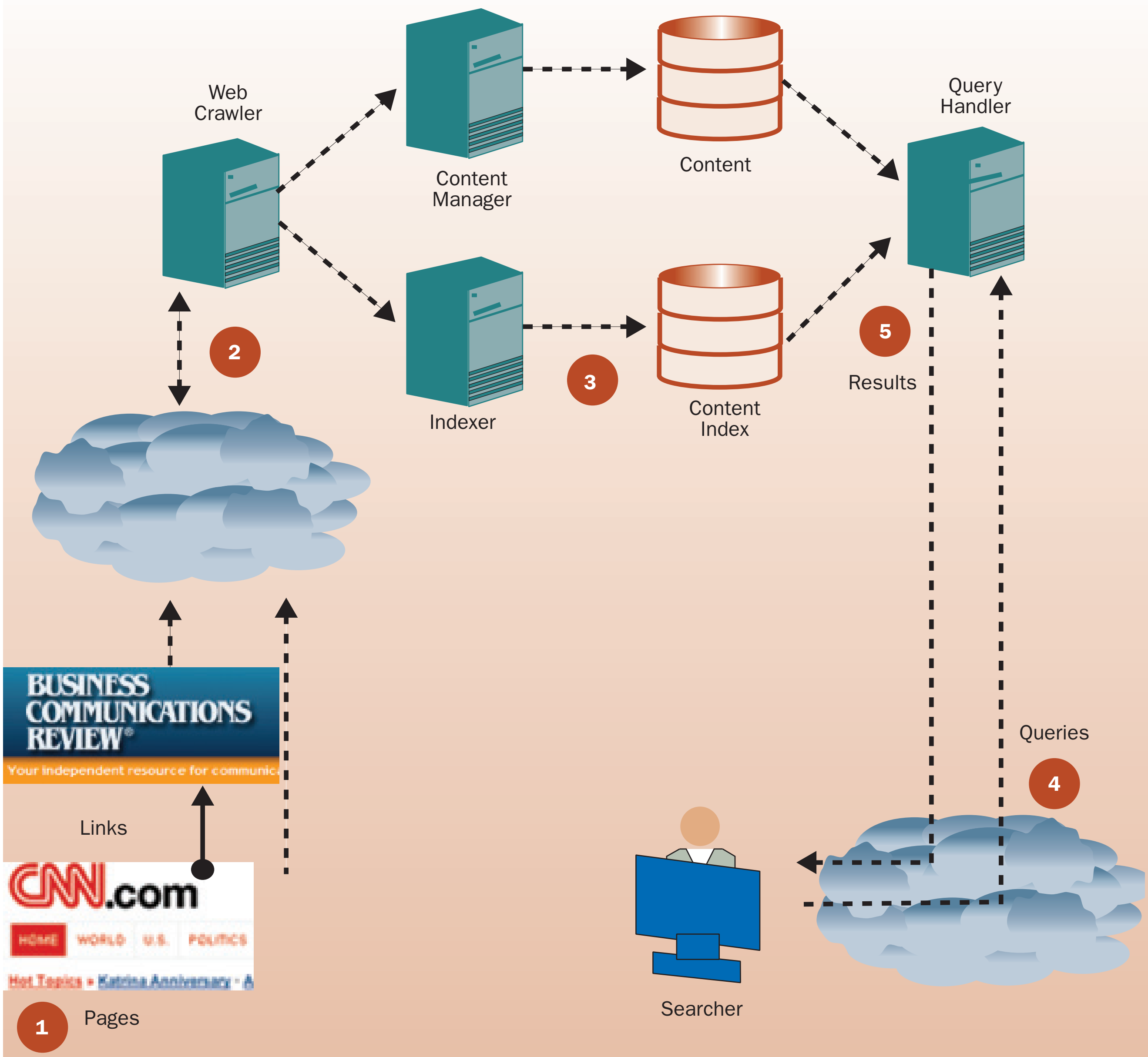
Link Analysis Search Engine Architecture

If you are reading this article, it's a safe bet that you've used an Internet search engine such as Google, Yahoo!, Live (MSN), or the like recently. In fact, you've probably used one several times already today. In many cases, exact details of search engine algorithms and systems are protected as trade secrets, and are continuously tweaked to optimize their behavior, so it's not surprising if you don't know exactly how they work.

At a high level though, Web searches are performed by a very straightforward process. As shown in Figure 1, first, a global community of Web page designers, document creators and multimedia editors create pages and other content and the hyperlinks between them. Second, a *Web crawler* or *spider* visits these pages, which it finds by recursively traversing hyperlinks from an initial root or list of pages. Third, an *indexer* notes each word on the page and keeps a record relating the URL to the word, and the word to the URL, in an inverted data structure called a *content index*. Typically, a *content manager* also maintains a thumbnail or compressed form of the content in a content repository. Fourth, searchers feed queries to a *query handler* or *query processor*, typically fed one or more keywords (although increasingly, query handlers may be fed images in addition to keywords [3] or find images related to a displayed image such as at Like.com [4]). Fifth, the query

Joe Weinman is responsible for emerging services and business development for a large global telecommunications service provider. He has a BS and MS in Computer Science from Cornell University and UW-Madison respectively, and has completed Executive Education at the International Institute for Management Development in Lausanne. He has been awarded 10 U.S. and international patents, and is a frequent industry speaker globally on strategy and emerging technologies. The views expressed are his own. He can be reached at joeweinman @gmail.com.

FIGURE 1 Basic Web Search



handler, by processing information in the content index and content repository, returns the exact URLs, thumbnails and/or snippets of text where those keywords may be found.

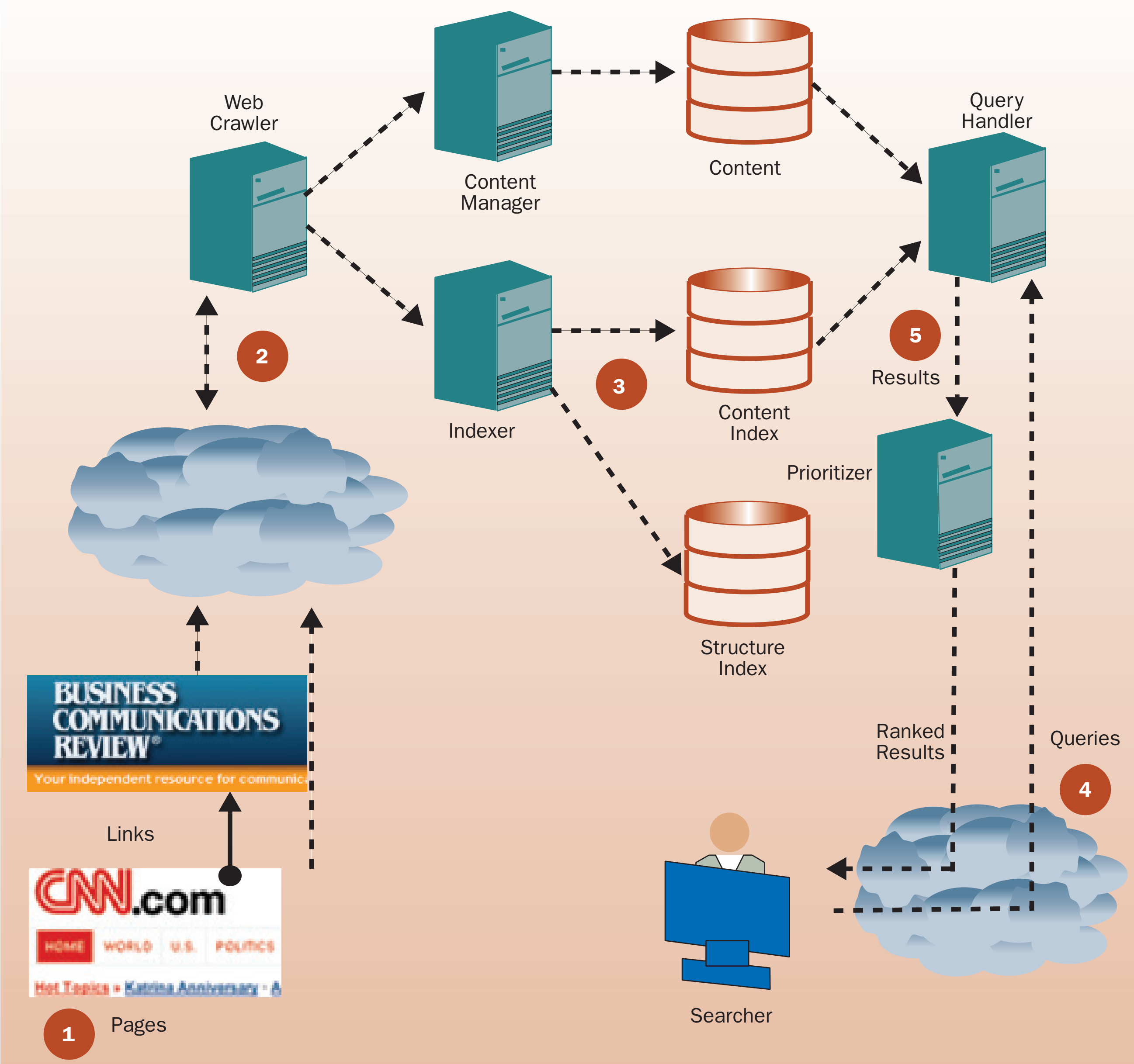
This is very much like inventorying the contents of a house. A crawler might walk through the house, visiting the kitchen, then the dining room, then the bedroom. The indexer would note that in the top drawer of the cabinet on the left of the kitchen it found notepaper, a pencil, a stapler and some chopsticks; in the bottom drawer of the dresser in the bedroom were some socks, loose change, a pencil and old Broadway tickets, and so forth. Then, a query handler, when asked if any pencils were in the house, might report back that, yes, there was one in the kitchen, and one in the bedroom dresser.

The problem is that, as the Internet has grown, simple queries can return millions of results. As of this writing, searching for "pencil" on Google is somewhat like asking for a list of where all the pencils are in the real world—the search returns 35.1 million results. The question then becomes not only where can I find "pencil," but also in what order should these locations be listed?

In other words, it is not only content-matching that is important, but also criteria such as relevance, authority, popularity, and/or importance. This is critical both to users, who want to get the most relevant results as quickly as possible, and to sellers of pencils, providers of pencil-oriented information, and so forth.

For paid advertising, there are a variety of means for determining ranking, including rele-

FIGURE 2 Implementing Algorithms Such As PageRank



vance, bid prices or hybrid schemes such as Yahoo!'s Panama algorithm. However, let's ignore the "paid results" schemes for now, and consider how best to objectively rank results to maximize user relevance.

The PageRank algorithm [5,6] is the essential idea behind Google search. The mathematics behind it includes such concepts as Markov Chains, Teleportation Vectors, Dominant Eigenvalues, Eigenvectors, Sub-stochastic Hyperlink Matrices, Transition Probabilities, Primitivity Adjustments, Quadratic Extrapolation and Convergence Criteria [2].

But the concept behind PageRank itself is very simple—so to understand how it works, we will use a simple example. Suppose that, instead of 35.1 million results searching for pencil, we got only two: a page at "Pencils Are Us," and another page titled "Pencils: What's the Point?" Which should be displayed first?

The PageRank algorithm might first ask: Which has more hyperlinks pointing to the page? For example, if Pen News Network, the International Pencil Association, and the "Save the Pencils" Foundation all link to the "What's the Point" article, and no one links to the "Pencils Are Us" article, PageRank would rate the oft-referenced page more highly.

What if both pages have, say, five references? Then, to determine priority, PageRank would consider each reference and determine which are themselves more cited. This recursive citation counting, or "backlink" approach lies at the heart of PageRank and related algorithms.

Figure 2 shows some additional components are added to implement an algorithm such as

PageRank. Specifically, the third step—besides managing compressed content encountered and (most importantly) managing the inverted index—also adds a *structure index*, which in turn feeds a *ranking module*, which we'll just call a prioritizer. The *prioritizer* takes the thousands or millions of results returned by the query handler, and sorts them in order based on their PageRank score.

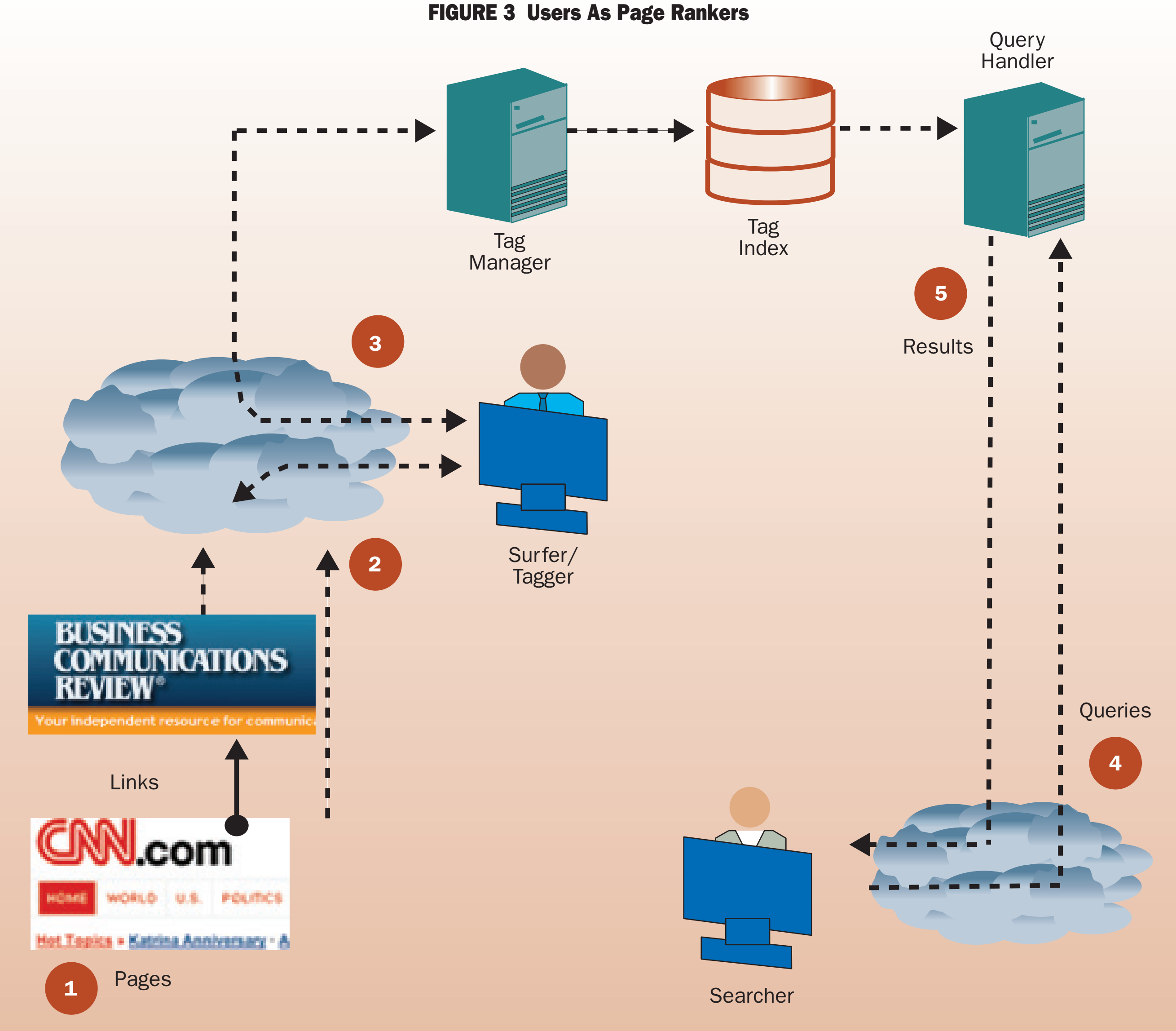
Other algorithms process hyperlink information differently; examples include HITS (Hypertext Induced Topic Search) [7], implemented by Teoma.com, which was acquired by Ask.com; and TrafficRank [8]. HITS recursively values sites as *authorities* (sites with many inbound links) and *hubs* (sites with many outbound links). The TrafficRank algorithm of Arasu, *et al.*, (not to be confused with Alexa.com's Traffic Rank) leverages the insight that each page has a number of inbound links and a number of outbound links. If each user arriving at a page must then leave the page, the recursive aggregation of these flows can help determine likelihood of visiting a page.

Fundamentally, however, both HITS and TrafficRank are in the same class of algorithms as PageRank, since these two also process hyperlink information mathematically to rank pages.

An issue with these approaches is that an unscrupulous pencil aficionado who wanted his page to rise to the top of the rankings might create lots of other pages that just pointed to that page. Such *link farms* are usually created on purpose, but occasionally arise innocuously when there is a high degree of internal linking, such as Wikipedia has. Many other tricks, such as cloaking, which returns quality results to Web crawlers but spam to users, and google-bombs, which temporarily enhance a site's ranking, are also used by unscrupulous search engine optimizers and pranksters.

Various adjustments therefore have been and will continue to be made to recursive citation, hubs and authorities, and inflow-outflow algo-

FIGURE 3 Users As Page Rankers



rithms. Such adjustments include considering the share of hyperlinks coming from a page or site. If a page only points to one other page, that is viewed as more of a vote than if it points to thousands of others as well. Other improvements, such as incorporating hyperlink anchor text in destination search results, building a scalable, high performance architecture, and so forth, are addressed in a classic paper [5].

Tagging, Voting, Recommendations And Collaborative Filtering

Other approaches aim to increase relevance by engaging volunteer surfers who actively contribute to databases oriented towards improving matching and relevance, and optionally providing recommendations of interest.

One strategy, used by sites such as del.icio.us, enlists active users to tag sites with keywords. These users, as shown in Figure 3, replace the automated Web crawler, but their objective is the same: develop an index of keywords used to map select keywords to Web resources. As before, the first step is the creation of linked pages, as occurs every millisecond on the Web. However, individuals, rather than an automated crawler, then surf the Web. As they encounter interesting content, they use a special Web page or a browser plug-in to tag the page with keywords, in effect using a tag manager to update a tag database. Then, when searchers look for pages matching those keywords, the query handler, leveraging the tag database, will serve up the appropriate URLs.

Figure 4 shows an enhanced strategy, where besides just indicating matching keywords, users also vote on the content found. Content may be Web pages, as in StumbleUpon.com or members of social networks, such as Spock.com and AfterVote.com. Variations combining tagging and voting exist both for Web pages at content sites as well as for surfing Web pages.

Figure 5 shows a simplified strategy, where content matching is ignored, and mere popularity is used. For example, Digg.com prioritizes news stories on its home page based on user votes, with no keyword matching at all. A story about politics

FIGURE 4 Users Vote On Content

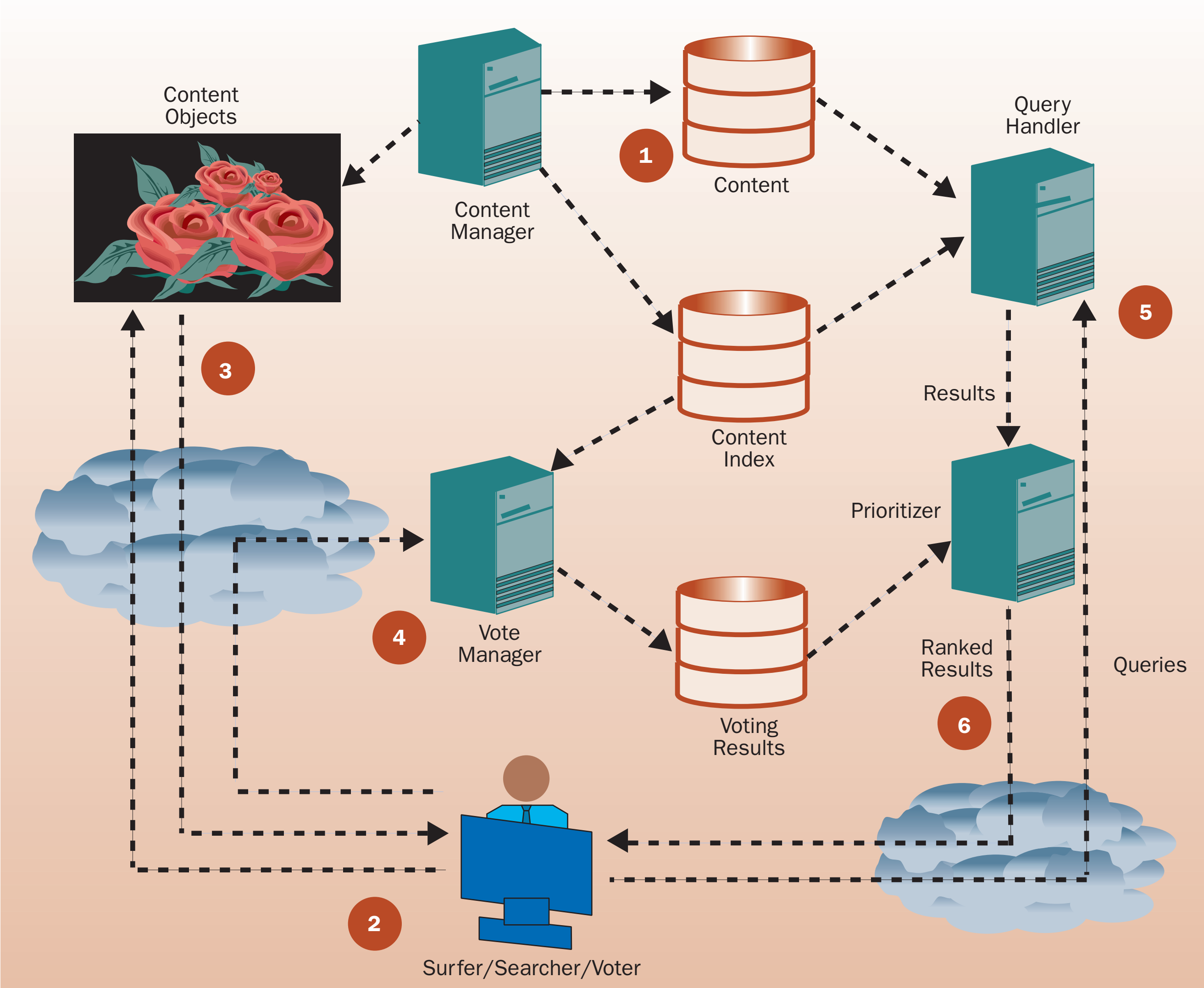
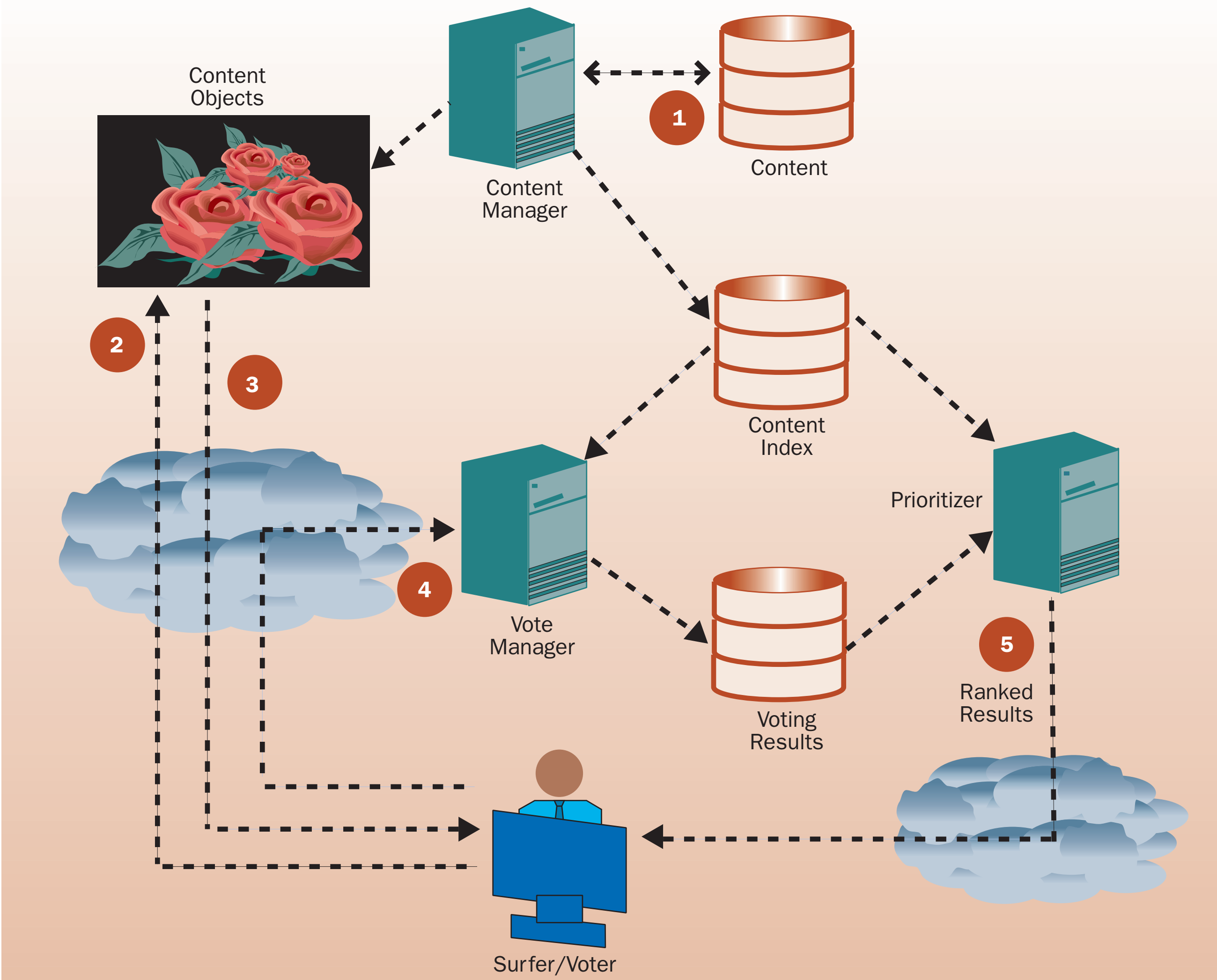


FIGURE 5 Simplified Strategy Using Popularity



might come right after a story about a lost cat. Truveo.com and Google Video offer most-popular lists. Even potentially off-line content, such as movies at NetFlix.com (which may be delivered in the physical mail as a DVD or over the Web as a streamed video) can be easily ranked in this way to determine the top movies.

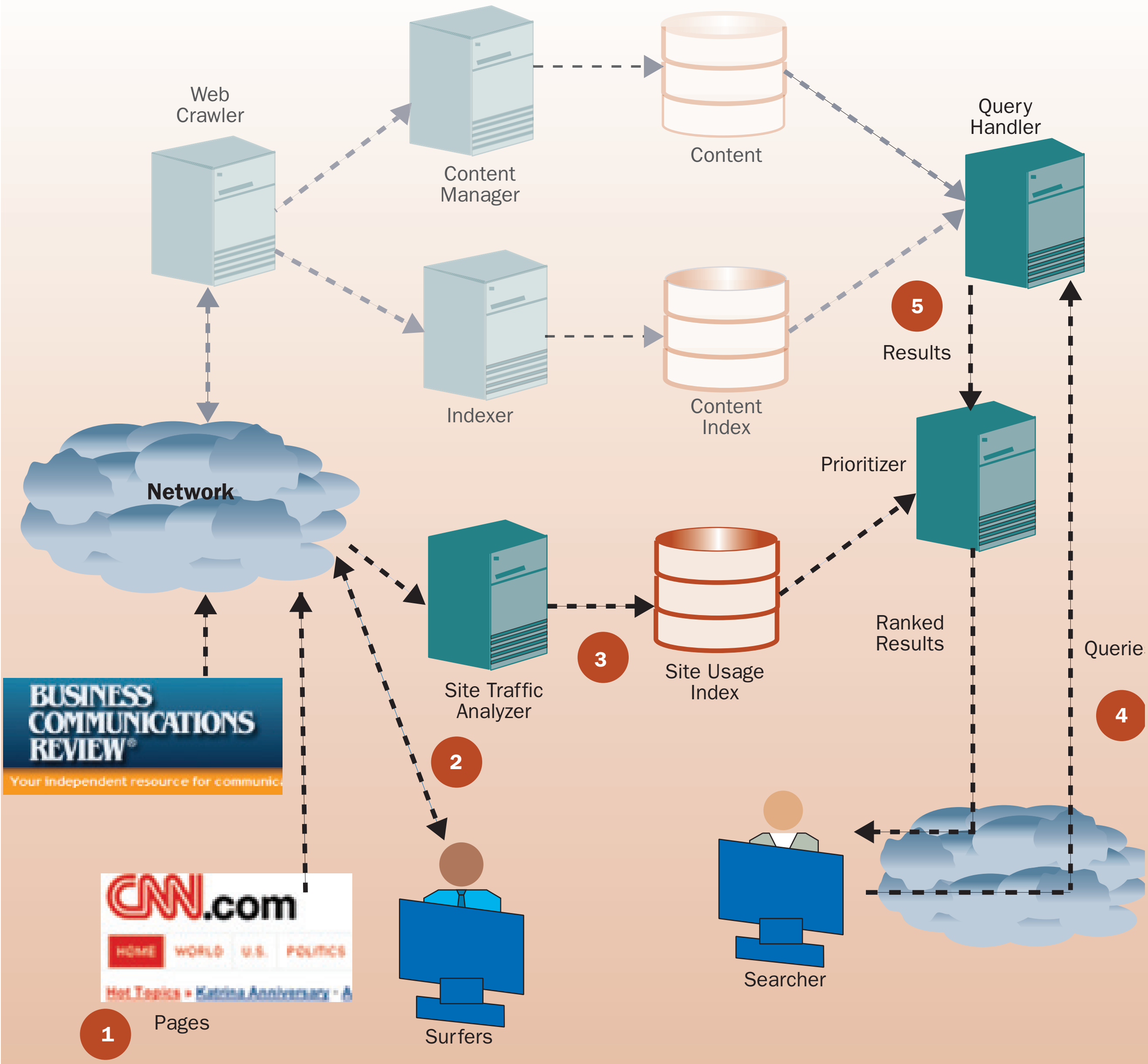
These same types of algorithms, which use hyperlink analysis, tagging, and/or voting, are in use directly or by meta-search engines (search results aggregators) across a wide variety of:

- **Internet search engines**—such as Technorati.com, ChaCha.com, Rollyo.com, Kosmix.com, Ask.com, Clusty.com, StumbleUpon.com, Draze MetaSearch, NetTrekker.com and Aftervote.com [9].
- **Enterprise search solutions**—from companies such as dtSearch, Google, IBM, ISYS Search Software, Mondosoft, Thunderstone Software, Vivisimo and X1 Technologies [10].
- **Enterprise bookmarking and tagging solutions**—from BEA, Cogenz, Connectbeam, IBM Lotus and the open source Scuttle [11].
- **Services providing recommendations**—such as Apple's iTunes, NetFlix.com and TiVo.
- **Mobile and location-based search engines**—which may use mobile browsers, SMS, a standard browser in a mobile laptop, or even voice as an interface. These include 4INFO, Earthcomber, go2, mDog, Slifter, UpSNAP [12] Yahoo! Local and Yahoo! oneSearch, and AT&T's YellowPages.com mobile and SMS search services. Since these search results may be returned in smaller form factors (for PDAs and cell phones), or over potentially limited bandwidth, correct results ranking is key.

This is even more important for services such as 4INFO and Yahoo! oneSearch, which return "answers, not links."

- **Intelligent engines**—as recently reported in *Information Week* [4], a sister publication to *BCR*. These include innovative approaches that use semantic and/or sentiment analysis, *e.g.*, Powerset, Hakia and IBM; queryless search, *e.g.*, MediaRiver, Blinkx.com and Yahoo!; and collaborative filtering, *e.g.*, Collarity,

FIGURE 6 Users' "Passive" Voting



The Issues With Legacy Approaches

The basic issue with traditional approaches—Internet search engines based on link analysis, enterprise search, bookmarking and tagging, mobile, location-based, recommendation-oriented, and/or intelligent—can best be explained by a quick analogy. Suppose you just arrived in a town you've never been in before, and find yourself in a tourist area with two restaurants. You'd like to have a bite to eat, and both menus have food that you love. One approach would be to count the number of reviews that each restaurant had received and, to break ties, consider the number of articles that had been written about those reviews.

Before heading to the library to dig up the reviews, one more observation might be helpful in making a decision.

One of the restaurants is *completely* empty. The maitre d' stands at the entrance, pleading with his eyes for you to come in and order something, anything…even a glass of water.

The other restaurant is packed, with happy diners waving champagne glasses in the air as they laugh and beckon to the waiters for more, and a line has formed—around the block!—of repeat customers waiting to get in.

Well, if you are really hungry, you *might* pick the first rather than wait, but if you were ranking the restaurants, you'd probably put the second one higher up on the list.

The lesson? That recursive weighted citation counting and other link analysis methods are interesting, but real-world traffic may be a much better criterion for ranking.

As for sites like Netflix.com, Digg.com or del.icio.us, another issue is that they require surfers or customers to take on an additional role: active tagging and/or voting. While many users are certainly willing to do this and even enjoy it, many are not, causing extra effort for and skewing results in favor of those who participate.

Some sites generate rankings passively: *e.g.*, the Netflix Top 100 Rental list or Dow Jones' WSJ.com "Most Viewed Stories" list. However, this information is restricted to the respective sites: Netflix does not also include Blockbuster's Top 100 list, and the Wall Street Journal does not include the New York Times list.

One approach to cross-site ranking based on passive data acquisition from users is Alexa.com, but this requires a browser plug-in, again, requiring an active step and with other issues as well.

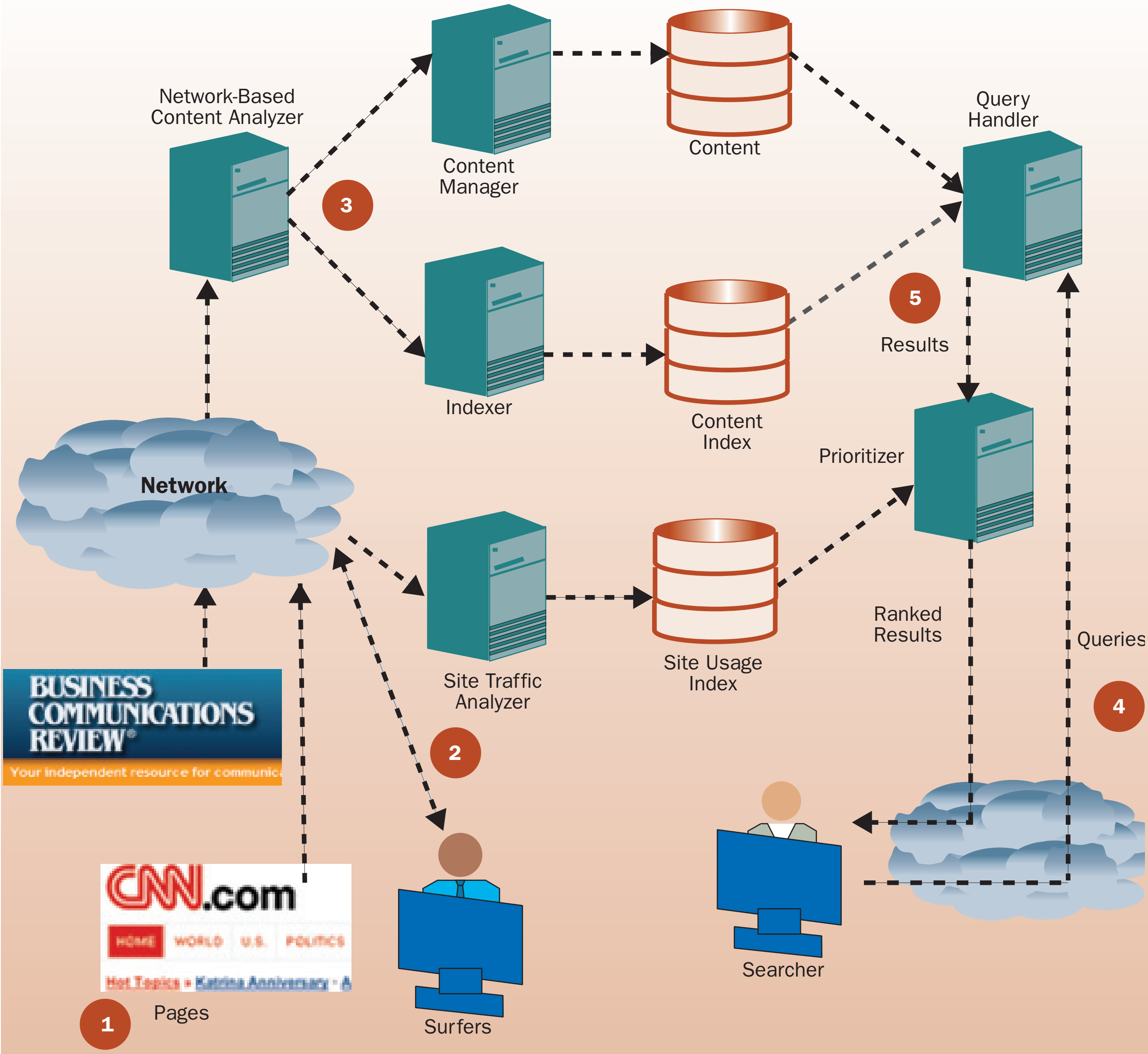
Rethinking Traffic-Weighted Search

In a World Wide Web context, the equivalent of festive diners lingering over their meals are surfers lingering over tasty morsels of Web content. Such data is easy to collect—for network service providers. Without intruding on anyone's privacy, anonymous statistical sampling of traffic delivered from websites could determine which sites and pages are more frequently accessed.

And not only could it be determined whether sites or pages are getting more traffic, but also whether diners are leaving after the first bite of their appetizer, or staying to enjoy the entire meal.

Consider a search for "International Pencil Graphite Producers" that turns up two results (actually, there are over 500,000 results. Scary.) One site, when visited, turns out to be a spam site for offshore pharmaceuticals, the other actually

FIGURE 7 Network-Based Content Analysis



has valuable information on graphite production and its impact on pencil use and the global economy. In general, when visitors go to the first site, their reaction is likely to be "Hey! This wasn't what I wanted!" and they will quickly click away or return to the search results page. When visitors go to the second site, they will tend to read the content, savoring the fascinating information regarding how graphite has transformed our society, clicking through to other in-depth pages regarding factoids about erasers and also why there is no lead in a "lead" pencil.

Consequently, network traffic statistics such as unique visitors, interval between visitor arrival at a page or site and departure from a page or site, packets transferred, subsequent clicks from a page vs. reloads of prior pages, clicks leading to other pages within a site, and similar types of measures could be an excellent indicator of average user interest in a page or site, which in turn is a proxy for relevance.

Moreover, if the data were collected by a network service provider on a statistically sampled basis and/or with outliers discarded, it would be very hard to spoof. A perpetrator of a non-useful site would have to convince a lot of users to spend their free time clicking around a given site, on the hope that random sampling would include a nontrivial fraction of these users in such a way as to artificially inflate the ranking of that site. While bot-nets could be enlisted to do something like this, better endpoint security and a method described below could ideally prevent it.

As shown in Figure 6, as surfers visit Web pages, they would, in effect, place passive relevance and usefulness votes which a *site traffic analyzer* would use to feed, maintain and update *site usage statistics*. This in turn would be used to tune query results ranking.

Rethinking Web Crawling

In fact, a network-based approach to search would apply not only to results ranking, but even to Web crawling. Rather than a Web crawler actively visiting sites to feed an indexer, a network service provider could automatically collect index information, and get that information on a more timely basis. For example, a Web crawler may visit a less popular site only once or twice a month. However, the first time a changed page was loaded by any user across a service provider's network backbone, that provider could passively detect that and update keyword indices.

As shown in Figure 7, a network-based content analyzer could take feeds from the service provider core IP backbone, and use it, with appropriate message assembly, to feed both a content repository and a content index. And this could work in conjunction with the site traffic analyzer described earlier.

The "Deep Web," or "Invisible Web," has been estimated [2] to be 500 times larger than the Surface Web accessible through hyperlinks. Although proportions have no doubt shifted due to online video sharing networks such as Google's YouTube.com, there is no doubt that there is still a massive amount of information hidden there. It comprises documents and databases such as patents, scientific papers, legal judgments and the like. It is available via the Web, but Web crawlers—searching based on hyperlink navigation—will not find it, because some information is not accessible via standard hyperlinks—only by, for example, entering information into a form which then drives form-data-based results. For example, even Google's own search engine will not, as of this writing, return results in the www.google.com/patents database. But, even though invisible to traditional hyperlink crawlers, it would be just as visible as any other content to a network-based content indexer.

In general, a traditional crawler approach could complement a network-based approach. Crawlers can find links to sites or pages that no one has already loaded, as long as these sites are referenced somewhere. Service providers could parse Web content, whether Deep or Surface, and be aware of changes well before a crawler, well, crawls over to it. After all, they are not called "Web Sprinters."

Which element is best suited to provide search results: Browsers, portals or network service providers?

Rethinking Click Fraud Detection

Another problem in today's search- and advertising-driven Internet is click fraud. Due to the prevalence of pay-per-click advertising and affiliate models on the Internet, there is plenty of opportunity for fraud.

For example, consider a site selling pencils. They advertise, and, when someone searches on pencils and clicks through to their site, the advertiser pays a certain amount. And they have ads placed through affiliate programs on other sites so that people visiting those sites, suddenly aware of their writing implement deficiencies, click through an affiliate ad to get to their site, hopefully to buy pencils.

However, this can lead to massive fraud. For example, a competing pencil vendor may click repeatedly through to the site, hoping to drive up advertising costs for the competitor. Or, an affiliate site may just click through to get a percentage of that advertising spend, never intending to purchase any pencils.

Here again, a network service provider could detect and ameliorate fraud. For example, detecting an IP address that repeatedly visits the same site through an affiliate referrer and then spends no time at that site would be an indicator that that endpoint is engaging in fraudulent behavior. On the other hand, a genuine pencil-purchaser would, once at the site, linger over the exciting images of writing implements for sale, and then click through to the "buy now" page. Depending on use policies, fraudulent users could be blocked, or

Network providers may look to search as a way of offering something beyond commodity services

advertising charges could be reduced proportionate to the percentage of fraudulent use.

Service Providers' Strategic Advantage

Who is in the best position to provide optimal search results rankings, Web crawling and click-fraud detection? There are three main candidates: client browsers, search portals and network service providers.

A number of issues arise with using browsers for ranking. Collecting data from them may be considered intrusive by some; additional traffic may be created as browsers report their activities, and there may be a skewing effect (do people who use IE have the same attitudes and orientation as ones who use Opera or FireFox?). These and other issues, such as sampling errors, disproportionate site correlations, global adoption differentials, definitions of "sites," and inaccuracies in data collection for sites with low traffic, have been documented [13]. Finally, at least for now, traffic data is only measured in terms of unique visitors and page views, and only at the granularity of the site, rather than measuring user interest and value at the granularity of each page in the site.

Search portals have a different problem. Once results are delivered, and the user clicks away, the search provider is out of the data path, and has no visibility into the traffic data until the user returns. If the user immediately reloads the same search page, that is certainly an indicator that the result they visited was not fruitful. But if they don't, did they find what they were looking for? Decide to use a competing portal? Take a lunch break? Get crushed by a meteorite?

Another issue with search providers is that many people bypass them for many types of results. For example, if, after reading this article, you decide to investigate pencils further, you might go to a search engine and type in "pencil."

Or, you might type in "pencil" into your location bar. Or you might type in pencil.com (which leads to an unexpected destination, actually).

A network service provider could best determine aggregate surfing behavior and hold times at sites or pages, in a way orthogonal to the peculiarities of browser preferences and regardless of whether a search engine is used. In general, the more backbone traffic carried, the more accurate the results, and the more regionally aligned the presence, the more accurate the results. For example, a Far Eastern backbone provider probably does not have a very strong ability to prioritize results for U.S. visitors visiting North American sites. While this issue will not impact, say, American users using an American search engine to search for English language results, it would certainly be a limitation in the usefulness and validity of any particular regional service provider or PTT attempting to provide one-stop shopping for enhanced search. Therefore, providers with a stronger global presence are likely to have a stronger base of aggregate data for ranking. One can also imagine a business model emerging for ranking based on data aggregated from multiple service providers.

And, size matters…to an extent. In determining rankings for frequently visited sites, a large backbone provider may not have much of an edge over a smaller one. But for more obscure pages and sites, a large backbone provider, especially one with a strong access business, has a statistically greater likelihood of carrying traffic to and from those sites, which might mean the difference between seeing traffic and not seeing it.

Yet another advantage of network service providers exists in Web crawling. Web crawlers add to the traffic on the Internet backbone as well as on Web servers themselves. Service provider indexing that bypasses Web crawling entirely can passively generate the same information, have greater currency (i.e., recency), and not have any impact whatsoever on Web servers.

Today's carrier routers are being enhanced in ways that would support implementation of the algorithms described here. First, routers are beginning to appear that are capable of application-layer message processing. Examples include Cisco's Application-Oriented Networking (AON), embodied as an appliance or a router blade, and Solace Systems' VRS/32 content-aware routing system, which can process messages at line speed.

In the same way that today's server-based crawlers use hyperlink anchor text as an additional data source for building a content index, a network-based crawler could ideally correlate HTTP GET and POST data with returned results to help index the Deep Web. Application-layer message and session processing would assist with this.

Other architectural techniques can complement existing carrier core and edge routers, by using passive optical splitters to feed traffic at line speed to an indexer. These passive techniques are invisible to core or edge routers, i.e., they do not add any processing load or latency to the routers.

One challenge in message assembly from packet flows could be due to load-balancing across access routers. Although a data analysis layer can reconstruct messages flowing across diverse paths, another solution exists: Virtual circuit switching through a technique such as AT&T's Intelligent Routing Service Control Point [14], which provides fine-grained route control to packet flows, rather than using load balancing in conjunction with protocols such as Open Shortest Path First (OSPF) or Enhanced Interior Gateway Routing Protocol (EIGRP), which can spray sequential packets across multiple equal-cost routes like a garden hose.

Conclusion

Today's Internet is a complex ecosystem of content providers, advertisers, portals, fixed and mobile endpoints, and network service providers.

Network service providers can be more than passive carriers of content, enhancing the ecosystem by providing better search results for users, and a more legitimate advertising payment model.

As network service providers seek to enhance their value-add, rather than risking commoditization by "over-the-top" content and application providers, they can leverage their unique strengths to enhance what is arguably the most important function on today's Web and intranets: search.

Rather than being an either/or situation, classic link analysis search results ranking engines based on the PageRank, HITS, or so-called "TrafficRank" algorithms could complement real-time, sampled, and smoothed network-based data to provide optimally relevant rankings to users. At the very least, a network-based approach could provide another data point to fine-tune search results rankings. Similarly, network-based indexing could complement traditional Web crawlers to better discover the Deep Web and improve the currency of index updates.

As the art and science of search continue to evolve, now is the time to consider the substantial role that the network can play□

References

1. Batelle, John, The Search: How Google and Its Rivals Rewrote the Rules of Business and Transformed Our Culture, Portfolio Hardcover, 2005.
2. Langville, Amy N. and Meyer, Carl D., Google's PageRank and Beyond: The Science of Search Engine Rankings, Princeton University Press, 2006.
3. Stix, Gary, "A Farewell to Keywords," Scientific American, July, 2006, pp. 91-93.
4. Hoover, J. Nicholas, "The Ultimate Answer Machine," Information Week, Aug. 6, 2007, pp. 41-47.
5. Brin, Sergey and Page, Lawrence, "The Anatomy of a Large-Scale Hypertextual Web Search Engine," available at http://infolab.stanford.edu/~backrub/google.html.
6. Page, Lawrence, "Method for Node Ranking in a Linked Database," United States Patent #6,285,999, Sep. 4, 2001.
7. Kleinberg, Jon, "Authoritative sources in a hyperlinked environment," Journal of the Association for Computing Machinery, 46, 1999.
8. Arasu, Arvind, Novak, Jasmine, Tomkins, Andrew, and Tomlin, John, "PageRank computation and the structure of the Web: Experiments and algorithms," The Eleventh International WWW Conference, Association of Computing Machinery, NY, May, 2002.
9. Monson, Kyle, "11 Ways to Search," PC Magazine, September 4, 2007, pp. 72-74.
10. DuPont, Ben, "Seek and Maybe You'll Find," Network Computing, June 25, 2007, pp. 53-58.
11. Greenfield, Dave, "Crowd Control," eWeek, August 13, 2007, pp. 36-43.
12. Flamig, Blaine, "On-The-Go Seeking: The Ever Expanding World of Mobile Search Tools," PC Today, September, 2007, pp. 18-20.
13. "About the Alexa Traffic Rankings," available at http://www.alexa.com/site/help/traffic_learn_more
14. Iloglu, Ali M., Nguyen, Han Q., Mulligan, John T., and D'Souza, Kevin L., "Method and apparatus for providing dynamic traffic control within a communications network," United States Patent Application 2006020606, September 14, 2006.

Companies Mentioned In This Article

4INFO.net
Aftervote.com (aftervote.com)
Alexa.com (alexa.com)
Apple (apple.com)
Ask.com (ask.com)
AT&T (att.com)
Autonomy (autonomy.com)
BEA (bea.com)
Blinkx.com (blinkx.com)
Cantfindongoogle.com (cantfindongoogle.com)
Chacha.com (chacha.com)
Cisco (cisco.com)
Clusty.com (clusty.com)
Cogenz (cogenz.com)
Collarity (collarity.com)
Connectbeam (connectbeam.com)
Del.icio.us (del.icio.us)
Digg.com (digg.com)
Draze.com (draze.com)
dtSearch (dtsearch.com)
Earthcomber (earthcomber.com
Endeca (endeca.com)
Fast Search and Transfer (fastsearch.com)
Go2.com (go2.com)
Google (google.com)
Hakia (hakia.com)
IBM (ibm.com)
ISYS Search Software (isys-search.com)
Kosmix.com (kosmix.com)
mDog.com (mdog.com)
MediaRiver (mediariver.com)
Microsoft (Microsoft.com)
Mondosoft (mondosoft.com)
Nettrekker.com (nettrekker.com)
Nexidia (nexidia.com)
Netflix (netflix.com)
Powerset (powerset.com)
Rollyo.com (rollyo.com)
Scuttle (scuttle.com)
Solace Systems (solacesystems.com)
Sonic Foundry (sonicfoundry.com)
Slifter.com (slifter.com)
Spock.com (spock.com)
Stumbleupon.com (stumbleupon.com)
Technorati.com (technorati.com)
Thunderstone Software (thunderstone.com)
TiVo (tivo.com)
Truveo.com (truveo.com)
UpSNAP.com (upsnap.com)
Veoh.com (veoh.com)
Vivisimo (vivismo.com)
Wikipedia.org (wikipedia.com)
X1 Technologies (x1.com)
Yahoo! (yahoo.com
YouTube (youtube.com)