Search
Engine Watch

How Search Engines Work

by Mike Grehan

Excerpted from the 2002 publication of
Search Engine Marketing: The Essential Best Practice Guide.

incisivemedia

Incisive Interactive Marketing LLC.
55 Broad St, 22nd Floor, New York, NY 10004

For more expert insight, visit us: www.searchenginewatch.com
Follow the author on Twitter: @mikegrehan

How Search Engines Work

by Mike Grehan

NOTE FROM THE AUTHOR

This year (2012) it will be ten years since I wrote the second edition of a book about search engines called *Search Engine Marketing: The Essential Best Practice Guide.*

I decided to revisit it recently. Writing it was very difficult because there was nowhere near the amount of information available about the inner workings of search engines and information retrieval on the web back in the day. So once I finished it, I breathed a sigh of relief and have very rarely ventured back into its pages.

Even now, I frequently meet people at conferences who bought it and still regard it as a useful resource. And surprisingly for me, having just re-read the most important parts of it, I also find a lot of it to be as relevant and fresh now as it was a decade ago.

I've been approached so many times over the years to write another book about search. And on a number of occasions with major publishers, in principle I've said yes. But then when I realize I'm just expected to write the same old thing that exists in any number of search marketing books (and there's a plethora of them available now with seemingly a new one published virtually every week) the wheels fall off, as it were.

The reason I wrote the book in the first place was because every other publication I read on the subject had a section called: How search engines work. And yet, not one of them actually explained anything about the application of information retrieval science or network theory, the principle drivers of search engine technology. Almost all of them had a cute "Incy Wincy Spider" type graphic with a row of filing cabinets and a brief explanation of how a search engine crawler works.

So, I embarked on a mission to at least document some of the history, theory and practical elements of what makes a search engine like Google tick. Bearing in mind, as I mention in the book, my background is as a marketer, not as a scientist. So although it demystifies some of the assumptions and anecdotal evidence of how search engines work at various forums and webmaster communities, it is a very simplified approach compared to that of an information retrieval practitioner or researcher.

It's likely, just reading this note that you'll be thinking: "Ten year old search engine stuff has got to be stale." But read on. No matter where you are in the industry as a search marketer, I do honestly believe there are one or two things that you'll hit upon which are actually new to you.

The following text is the chapter of the book on how search engines work and I'm leaving it entirely in the quirky, very British way it was written. The only changes are, I removed an entire section about themed web sites (a controversial topic at the time) and a section about a research paper called the term vector database (an even more controversial topic at the time). Neither topic has any real relevance ten years later.

About the Author

Mike Grehan is VP, global content director, with Incisive Media, publisher of Search Engine Watch and ClickZ, and producer of the SES international conference series. He was elected to SEMPO's board of directors in March 2010.

Formerly, Mike worked as a search marketing consultant with a number of international agencies, handling such global clients as SAP and Motorola. Recognized as a leading search marketing expert, Mike came online in 1995 and is author of numerous books and white papers on the subject. He is also chair of the SES Global Advisory Board and is currently writing his third book due in summer 2012.

Also, I make reference to some search engine experts during the chapter. This is because I was fortunate enough to talk to them during the research period. So that you can keep everything in context, I've included the verbatim transcripts of my interviews.

And just in case you're curious, yes I will be publishing a new book next year. It's called Connected Marketing: Reaching The Transient Media Consumer. And no – it's absolutely not an SEO book!

Cheers!

Mike Grehan

CONTENTS

OVERVIEW

In the first edition of this guide I explained the way that search engines differ from directories and how they go about the process of collecting and building their own individual indexes and use unique retrieval systems to rank results following user queries. I'd like to elaborate on the subject this time and take a more in-depth look at exactly how crawler based search engines work, and as it's very important, also differentiate the categories of search as the search engines see it. Once you understand what it is that search engines themselves are trying to achieve and how they go about it, it will be easier to understand the results as you see them appear on the page following a keyword search. This will help you to rationalise and then optimise the way that you create web pages to be indexed and gain a better understanding of why it's essential to do it. I should mention here that, some aspects of this section are of a highly technical or scientific nature. I have tried to keep it to the fundamentals but also included as much background as possible should you wish to continue further research yourself into the subject.

A search engine can only search the subset of the web which it has 'captured' and included in its own database.

As it's not true that search engines actually 'search the web' once you have keyed in your query, even though they all usually have a 'caption' of that type next to the search box, you will always get different results in different places. It's pure myth that search engines scan the whole web looking for matching pages following a query. A search engine can only search the subset of the web which it has 'captured' and included in its own database. And of course, the amount of content and the 'freshness' of the data relies solely on how often that database is updated/refreshed.

The largest search engines are index based in a similar manner to that of a library. Having stored a large fraction of the web in massive indices, they then need to quickly return relevant documents against a given keyword or phrase. But the variation of web pages, in terms of composition, quality and content, is even greater than the scale of the raw data itself. The web as a whole has no unifying structure, with an enormous variant in the style of authoring and content far wider and more complex than in traditional collections of text documents. This makes it almost impossible for a search engine to apply conventional techniques used in database management and information retrieval.

As is also mentioned in the section on how directories work, we tend to use the term 'search engine' generically for all search services on the web. It's also interesting to note that, when the term is used with regard to the crawler based search engines, they tend to be referred to as though they were all the same thing. The fact of the matter is, even though they all, in the main, use spiders/robots to find content to build their database, they all collect and record different amounts and different types of information to index. And following a keyword search, they all retrieve the information from their unique databases in different ways.

The retrieval algorithms (mathematical computer programming methods which sort and rank search results) which each of the major search services use for ranking purposes are also quite unique to each specific service. Prove this yourself by typing in a keyword or phrase into the search box at Google and note the results. Then go to Alta Vista and repeat the exercise. You'll always find different results at different search engines. Some search services provide duplicate results i.e. at the time of writing there are 'cloned' results from Overture (formerly GoTo) at both NBCi (formerly Snap) and Go (formerly Infoseek). Although these particular services can largely be regarded as defunct, they still remain online. So (in the main) even if pages from your site are indexed with all of the major search services, you're most likely to have a different rank with each individual service.

Google, as the world's biggest search engine, in the sense of both its popularity amongst surfers and its database of almost three billion files (Google's own reported figure - 2001), still only has proportionately a small amount of what's actually on the web. The web grows exponentially. Research carried out in 2000 discovered an estimated 7.5 million pages being added every day [Lyman, Varian et al – 2000] so it's not really feasible that any search engine will ever have the whole of the web sitting around on its hard drive being refreshed every day to keep it completely current!

The practical constraints alone are a major problem i.e. the size of a web page has been gauged at an average of about 5-10K bytes of text, so even just the textual data which a large search engine records, is already into the tens of terabytes when it comes to storage.

And then there is what's known as the 'invisible web' i.e. more than 550 billion documents [Lyman, Varian et al - 2000] that the search engines are either not aware of (not submitted to them and not yet linked to by any other pages), choose to ignore or cannot access, (some dynamically delivered content sites and password protected sites), or their current technology simply does not yet enable them to capture (pages which only include difficult file types like audio visual, animation, executable, compressed etc.). To continually crawl the web to index and re-index as many documents as they already do is not an inexpensive task, as you will see when we go through the anatomy of a search engine one step at a time. Implementing and maintaining a search engine database is an intensive operation which requires a lot of investment to provide the necessary technical resources and continued research and development.

So, even if we do use the term 'search engines' generically as though they were all the same thing with different names, the probability is that they all actually vary enormously in how comprehensive and current they are. Google may have more pages indexed than, say, Fast (www.alltheweb.com), but if Fast has updated it's index more recently than Google, or recently crawled a newer subset of the web, then even with less pages it may return fresher and more comprehensive results at certain times. There are also many other factors beyond the basic technical process of the crawler module used by search engines which need to be taken into account.

'Off the page' criteria or 'heuristics' play an enormous part in the different ways that crawler based services operate.

I should mention here that, search engines frequently quote the sheer volume of pages held in their database as an indication of being the best or the most comprehensive. Of course, the frequent trade-off between quantity and quality is very much at play here. Although size is clearly an important indicator, other measures relating to the quality of the database may provide a better insight as to just how relevant their results are following a keyword search. Finding 'important' relevant pages on the web for indexing is a priority for search engines. But how can a machine know which are 'important' pages? Later in this section I will explain some of the methods used by search engines to determine what makes certain web pages more important than others.

Because search engines frequently return irrelevant results to queries, I should also expand a little more on one of the many problems they have in attempting to keep their databases fresh. Aside from new pages being added to the web, older pages are continually updated. As an indication, in an academic study of half a million pages over four months, it was estimated that over 23% of all web pages were updated on a daily basis (in the .com domain alone over 40% of pages were changed daily) and the half-life of pages was about ten days (in ten days half the pages are gone i.e. a specific URL is no longer valid) [Arasu, Cho, Garcia-Molina et al – 2001]

Search engine Spiders find millions of pages a day to be fired back to their repository and index modules. But as you'll gather from the above, the frequency of changes to pages is very hard for them to determine. A search engine spider can crawl a page once, then return to refresh the page at a later stage and it may be able to detect that there have been changes made. But it cannot detect how many times the page has changed since the last visit. Certain web sites change on a very frequent basis i.e. news web sites or e-commerce sites which have special promotions and price changes etc. Much research work is continuously carried out in both academic and commercial sectors to develop and devise 'training' techniques and other methods for crawlers. But even if an 'important' page can be crawled every 48 hours, there is still room for human intervention from webmasters which happens on a very large scale.

> Finding 'important' relevant pages on the web for indexing is a priority for search engines. But how can a machine know which are 'important' pages?

If a webmaster uploads a page to the server and then either submits the page via the 'Submit URL' page at a search engine, or if the page is simply found by a search engine via a link from another site (as is more likely) during a crawl, the content and composition of the page as it was crawled is what will be indexed. So, if on the first day of indexing, the page has a particular number of words which are contained in a specific number of paragraphs and a certain keyword ratio or density, then this is how it will be recorded and this is how it will remain indexed until the next time it is crawled. If the author of the page then decides to make new additions like images and captions and edits the text, the search engine will not be aware of this until its next visit.

If a surfer performs a query for the specific topic of the page content on day one of the search engine indexing and updating, then the page will be returned with the relevant information as recorded. However, if they perform the search after the author has changed the page, the search engine will return the page against the same keyword/phrase used, even though the author may have altered the context or taken out important references to the topic without making the search engines aware of it. This then presents the surfer with the frustration of not getting a relevant page to go with the query.

This, as you can see, is a major problem for search engines – they simply cannot keep up with the growth of the web and the constant changes which are being made to web pages. The 'crawler lag' issue can be as short as 48 hours with 'pay for inclusion' programs like the one provided by Position Technologies on behalf of Inktomi, or as long as 4-6 weeks (sometimes even longer) for a basic submit (Google claims to refresh tens of millions of 'important' pages on a daily basis, but this is still a tiny subset of the web). So, again, even if on the outside search engines look to be the same thing or similar – what you see in their results to a query actually all depends on the parts of the web they have managed to index to date, how fresh the data is, external influences and then how they choose to rank and return the results to the user.

There is also (as is mentioned in the 'How Directories Work' section) a 'grey' area in the pure definition of the term search engine, because even crawler based search engines provide and return directory results, i.e. Google provides and returns results from Open Directory in its mix and Yahoo! licenses and returns results from crawler based Google in its mix (although I don't wish to confuse the issue any further as this is covered in more detail in the Submitting section, I should also mention that Looksmart returns results from Inktomi [soon to be Wisenut at the time of writing] in its mix). For the directories these are secondary results which occur when they don't find a specific match in their own listings (also known as 'fall out' or 'fall through' results). All the more, even though I try hard to help differentiate between the crawler based services and the directories, they do tend to merge in certain places and therefore the finer the dividing line of differentiation becomes for the casual surfer.

> This, as you can see, is a major problem for search engines – they simply cannot keep up with the growth of the web and the constant changes which are being made to web pages.

Perhaps it's more correct to say that, most of the major search services could now really be viewed more as hybrids. And it's not just for the benefit of the surfer that a crawler based service provides directory listings and that a directory uses crawler type algorithms for ranking purposes. It's the luxuries they afford each other. Google can't afford to have editors sifting through billions of pages to give them a personal quality check. And Yahoo! can't depend on all of its users wanting to drill down through hundreds of categories to find the information they're looking for. So it makes sense for Google and others to build a bit of priority into their results for those pages which they know an editor from Yahoo!, Looksmart or ODP (preferably all three) has physically visited to qualify them.

And it makes sense for the directories to adopt the retrieval technologies used by the crawlers as well as presenting secondary results which help to overcome the limitations of their much smaller databases (NB there is an inherent flaw in using the search box at directories and this is covered in that section).

Then there are the databases behind the databases. Let's take Microsoft for instance (I know many who would like to take Microsoft – and dump it into the sea!). On the surface, when you go to [www.msn.com] you may get the impression that you've arrived at the Microsoft search engine service. To all intents and purposes, it is. But Microsoft does not crawl the web looking for sites to populate its own database (at this time). They actually rank and return a combination of results from other sources. They license access to the Looksmart Directory and Inktomi database then use their own retrieval and ranking technology for the main body of results, but for their top of the pile they use results from Overture's database. The same process applies with HotBot which pulls in results from Inktomi, ODP and at the top of the pile from Direct Hit. AOL search pulls them in from Inktomi, ODP and at the top of the pile Overture. [NB: Check the section on major players which had any amendments to the above made on the day this edition was published]

These 'top of the pile' results mentioned above, add yet another confusing dimension, in that, many of the major search engines and directories, like Yahoo!, which at the time of writing had entered into a deal with Overture to provide 'top of the pile' results, share their resources with other online search services on a commercial basis. This is yet another flaw which affects the results that appear at the top of the page with all of the search services – the results which appear at the top may not be the most current (perhaps not even most relevant) – the web site owner just paid the most money to appear there. Just about every search service online provides some form of 'paid for' listings at the top of the pile (see Pay Per Click section).

The term 'portal' is also frequently interchangeable with 'search engine' for surfers in many cases. A number of the major search services have integrated portal features into the home page of their sites (Google never did and Alta Vista dropped the idea as a business model) and almost all true portals i.e. what are really destination sites like www.canada.com include a 'search the web' box amongst the other clutter on their home pages (search results for www.canada.com are supplied by meta search engine Dogpile). By this, I mean that they present you with news feeds, entertainment and financial information as well as email and messenger services etc. The intention here is simply tactical for what are really just online brands, to lure you into making their home page - your home page, i.e. the first page you see when you open your browser and go online (brand loyalty tactic). It's possible, with some of the search portals to be able to configure the presentation of the page to suit the users' own preferences. For instance, at Lycos you can personalise the page content, colour and the way that the information is presented by having news fed through first, or sports news more prominently etc.

All of this can be quite confusing to anyone new to search engines and the process of search engine optimisation. But once you are able to understand where search results are going to and coming from with the various major search services, you can concentrate on targeting just the most important ones and those which you are likely to be able to have some kind of positive influence over with your optimisation efforts. The intention of this guide, of course, is to help you to unravel the whole tangled mess (as it appears to be) and then help you to make some sense of it!

THE CHARACTERISTICS OF SEARCH FROM A SEARCH ENGINE POINT OF VIEW. (OR TRYING TO EMULATE THE "LITTLE OLD LADY IN THE LIBRARY")

One thing which has come to the forefront in the research carried out by search engines in order to be able to provide more relevant results is the fact that conventional methods of information retrieval (IR) simply do not 'stand up' as well on the web. From the pioneering work in automatic text retrieval by the late Gerard Salton with the Vector Space Model (described later in this section) to the latest experimentation and developments of link analysis and machine learning techniques for text classification labelling (also described later), the question still remains: How do we get the results to be as effective and relevant as 'the little old lady librarian'?

> The results which appear at the top may not be the most current (perhaps not even most relevant) – the web site owner just paid the most money to appear there. Just about every search service online provides some form of 'paid for' listings at the top of the pile

In most of the conversations I've had with leading industry figures like Andrei Broder (Chief Scientist – Alta Vista), Craig Silverstein (Director Technology – Google) and innovators like Brian Pinkerton (Web Crawler – the web's first full text retrieval search engine), the analogy of the 'little old lady librarian' arises. And it's not just the crawler based search engines either, even Yahoo! used the analogy. So what is it that she does that search engines can't do? Well, let's take a look at the characterisation of search as seen by a search engine and then we'll come back to our "little old librarian".

Andrei Broder explained to me the basic characteristics of search, as he sees it, which fall into three broad classifications (I have no reason to believe that his explanation would not apply to all search engines). The first thing which Andrei was keen to point out was the difference between classical information retrieval and the problem it poses with the web. Although algorithms have been developed for traditional information retrieval to address hypertext systems, the web lacks the explicit structure and strong typing of these closed systems. Smaller, well controlled homogonous collections, such as scientific papers or news stories, for instance, are easier to retrieve and rank against set criteria. The Text Retrieval Conference (TREC) has described the benchmark for a very large corpus (body or collection of writings, text etc.) as being 100 gigabytes of information (Google already has tens of terabytes of information stored – to give an indication of size here, one entire copy of The Encyclopaedia Britannica would be about 1 gigabyte, a public library with over 300,000 books would be about 3 terabytes of information).

The question still remains: How do we get the results to be as effective and relevant as 'the little old lady librarian'?

The web, as we know, is a vast collection of heterogeneous pages developed in an uncontrolled manner by anyone with access which exceeds any corpus ever imagined before.

This lack of a governed structure or standard on the web has lead to the explosion of available information, but it also causes immense information retrieval problems for web search engines. The major problems being context and relevancy – just how relevant are the results we get?

These are the three [wide] classes to web searching as Andrei described (quotes lifted directly from the interview section of the guide):

Informational

Navigational

Transactional

(1) Informational.

"This applies to the surfer who is really looking for factual information on the web. So they make a query like say...'low haemoglobin' for instance. This is a medical condition. They are looking for specific information about this condition. That's very close to classical information retrieval."

(2) Navigational.

"Navigational is when a surfer really wants to reach a particular web site. If they do a query like, say, United Airlines, for instance. Probably what they really want is to go directly to the web site of United Airlines – like www.ua.com just like if someone typed BBC, it's most likely they want the web site of the BBC - and not the history of the BBC and broadcasting. They probably want to just go directly to the web site. We all do a lot of these types of searches, in fact, this accounts for about 20% of queries at Alta Vista".

(3) Transactional.

"Transactional means that ultimately the surfer wants to do something on the web, through the web. Shopping is a good example. You really want to buy stuff. Or you want to download a file, or find a service like, say, yellow pages. What you really want to do is get involved in a transaction of information or services. Take a shopping query, these are transactional queries where people want to buy stuff and so on. So, they are wanting a return which satisfies this need."

"So, I think it's important when you're talking about relevance and precision to distinguish between these three classes. Because, for instance, for the classic transactional query, with me living in California, it's likely to be something different to what you want living in the UK.

So what's happening with transactional queries, it's difficult to decide what the best result is. The context plays a big role.

And in fact, often with this type of transactional query, the traffic from other sources, is often better than what we collect ourselves. It's often more up to date or it's more appropriate because it's a pure shopping query, you know when you go shopping, you'd better be in a shopping mall – not in a library" [Laughs].

And at the mention of the word library... let's bring our 'little old lady librarian' back into the picture. You'll note from the above, as Andrei points out, it's just so difficult for a machine to be able to comprehend the nature of a query. It can bring back what it deems to be the most relevant documents by the keywords in the query, or in the link anchor text, even decide on citation and reputation (covered later), but it can't intuitively decide the nature or classification of the purpose of the search. Whereas, if you go into a library in a small town and walk up to the 'little old lady librarian' she can intuitively make some assumptions about the nature of your search and exactly where you'll find the appropriate texts.

As I've already explained, a great deal of what search engines attempt to achieve is based upon conventional information retrieval (IR) systems and procedures. Let's assume, for instance, that I go to the small library and ask for a specific and popular book. If the librarian realises that she does not have a copy of the book and perhaps, this is not the first request for it, then it's likely she will go out and find/order it. When she receives the book, she takes out some index cards and then records details of it. A note is made of the title of the book, the author's name, some key words describing the content, an identifying order number (ISBN), a category heading and index number for retrieval purposes. The book would then be placed on the repository/library shelves in alphabetical order and the index cards would be filed.

A good library system allows you to cross index items so that they can be found not just by title, but by the author name or even by category etc. By receiving many enquiries about a particular book or subject, a librarian can frequently, intuitively point to the exact book or at least to the category section. By checking books in and out of the library she can also make a note of popularity and usage i.e. how many times a popular book has been checked out, how many users checked it out more than once and, of course, how many books seem to appear to sit on the shelves doing nothing but gather dust. All of this information is useful in keeping the library up to date, with 'stale' books that are no longer popular being moved to a remote repository, making room for fresh new material or further copies of popular classics.

If you think about the two paragraphs that you've just read, you'll see that, in a very 'quaint' way, it is actually a description of the principle workings of a search engine. It appears to be such a simple and straight forward process, but the problems encountered by the most advanced computer technologies are vast when trying to emulate it.

In Andrei Broder's analogy, he gives an example of a schoolboy who may walk into the library and ask for a book about Italy. Here, the librarian, with scant information herself, can make an assumption that he may be writing an end of term paper for instance, and therefore he would need books about the history and culture of Italy. If a grown man walks into the library during the Summer and asks for information about Italy, she may assume (or determine in seconds) that he's going to Italy for a holiday and therefore he would need travel guides and point him to those texts.

Brian Pinkerton uses the same type of analogy when he says: "If you type a query into a search engine on a subject like Bora Bora, how would the search engine know whether to give you pages on the history of Bora Bora or pages about travelling to Bora Bora". Simply put, the librarian can help you focus towards more relevant topics and texts by understanding the nature and context of the search. This proves less of a problem for directories which are classified by design and this is why Frazer Lee made reference to the editors at Yahoo! being "the librarians" of the Internet. This brings us back to the original trade off between search engines and directories. Directories may have the upper-hand in a search because their users can drill down through a series of categories to get to relevant material only (in the main – a lot of esoteric categories simply don't exist). But because their index will always be much smaller than that of a search engine like Google, you will always have less (possibly even older) information even in those specific categories. Google or Alta Vista may not be able to determine the exact nature of your search, but they can certainly try to return what they deem to be specific and related pages as judged by the 'topology of the web' (covered later).

> A good library system allows you to cross index items so that they can be found not just by title, but by the author name or even by category etc. By receiving many enquiries about a particular book or subject, a librarian can frequently, intuitively point to the exact book or at least to the category section.

I mentioned to Andrei that, regardless of the nature of the query, most search engines still return anything from a few thousand results to a few million, of which only a fraction could really be relevant. So how quickly does the relevancy factor drop? Well, the answer is that, after the first couple of pages in the results, the relevancy factor begins to drop like a stone.

THE ANATOMY OF A SEARCH ENGINE (SEARCH ENGINES 'UNDER THE HOOD' – THAT'S 'BONNET' FOR UK READERS!).

Authors, researchers (and most certainly search engine optimisers) have tried to break down the components of a search engine in an effort to make it easier to explain what the process from crawling the web to returning results actually is. A good search engine working at its optimum performance should be able to provide effective and efficient location of web pages, thorough web coverage, fresh information, unbiased access to all material, an easy to use interface for surfers (which can handle basic or advanced queries) and the most relevant results for that moment in time.

I've mentioned elsewhere in this guide that, in real terms, the process of search engine optimisation can hardly be regarded as 'rocket science'. But the process of designing and implementing a search engine for the world wide web requires the skill and technology employed by qualified information retrieval and computer scientists.

Let's not forget that, for example, Larry Page and Sergey Brin (founders of Google), met as PhD candidates at Stanford University (as did both Jerry Yang and David Filo of Yahoo!). If you've looked at the section of the guide which covers the brief history of search engines, you'll be aware that most of the major search services started as university projects. And all of the major online search services employ computer and information retrieval scientists to further develop their technologies for the future. Trying to simplify what some of the world's leading experts in information retrieval and computer technology are pushing-to-the-limit, is not an easy task.

If you've looked at the section of the guide which covers the brief history of search engines, you'll be aware that most of the major search services started as university projects.

Providing content-based access to large quantities of text is a difficult task as you may have already gathered, and if not, will certainly now discover. Even with mountains of research we still have a poor understanding of the formal semantics of human language. As you will also discover here, the most successful methods and approaches to information retrieval, routing and categorisation of documents relies very much on statistical techniques.

As you may have read elsewhere in this guide, my background is in media and marketing (I'm not a mathematician or scientist), so I thought it might be wise, for the benefit of readers who may be more authoritative in the science of information retrieval and search technologies (and in advance of the next section) for me to humbly quote a great mathematician:

"When you can measure what you are speaking about, and express it in numbers, you know something about it; but when you cannot express it in numbers, your knowledge is of a meagre and unsatisfactory kind; it may be the beginning of knowledge, but you have scarcely in your thoughts advanced to the state of science."

William Thomson - Lord Kelvin.

I've decided to break down the anatomy and process into five essentials (there are many variants), hopefully this will help you to have a good fundamental understanding of how search engines operate technically.

On a less complicated level, search engines could simply be described as suites of computer programmes interacting and communicating with each other. Different, or various terms for the particular components are used by search engines in their development and research, but I have used basic terms and hopefully these explanations and descriptions are easier to grasp than those in more technical and scientific papers.

The crawler/Spider module.
The repository/database module.
The indexer/link analysis module.
The retrieval/ranking module.
The user query interface.
The crawler/Spider module:

(The terms crawler, spider and robot are used interchangeably here.)

Search engines keep their methods for crawling and ranking web pages very much as trade secrets. Each search engine has its own unique system. Although the algorithms they use may differ from one search engine to another, there are many practical similarities in the way they go about building their indices.

Search engine spiders discover web pages in three ways:

By using a starting 'seed set' of URL's (known web pages) and extracting links from them to follow (just pull them out of Yahoo! for instance).

From a list of URL's obtained by a previous crawl of the web (following the first results of the initial crawl).

Human input from webmasters by adding URL's directly at the search engine (now very much regarded as 'other input').

There are many complications encountered by search engine spiders due to the size of the web, it's continual growth and changing environment. As you are now aware, unlike traditional information retrieval, where all of the data is conveniently stored in a single location ready to be indexed - the information on the web is distributed over millions of web servers. This means that the information has to be gathered first and then systematically placed in large repositories before being passed on for processing/indexing. The design of a good web crawler should ensure that it is able to avoid the external problems it presents to the owners of web sites which can be 'bombarded', and also to be able to internally handle massive amounts of data: This certainly presents a challenge.

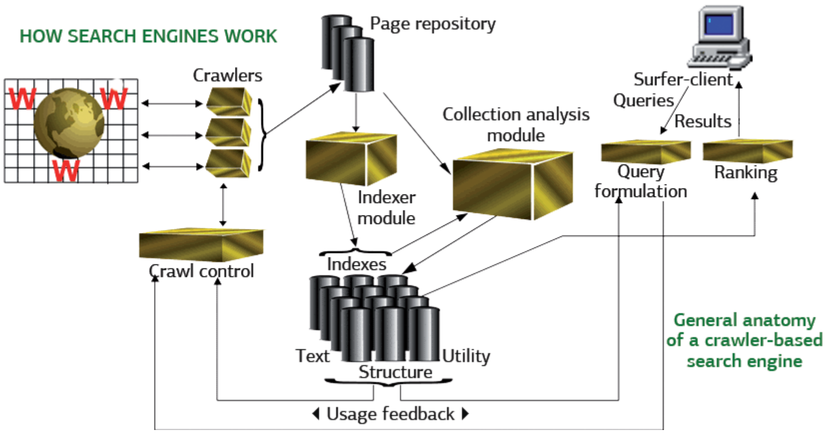
Both capacity in terms of resources and time, mean that it must be programmed to carefully decide which URL's (Uniform Resource Locator's – web page addresses) to scan, in what order and how frequently to revisit those pages.

Although this guide focuses only on the design and implementation of general purpose search engine crawlers, there are many different types of crawlers on the web. There are those which are used for personal use directly from a PC desktop, such those for harvesting e-mail addresses (e.g. Web Weasel) or other commercial spiders which are carrying out research, sizing the web and spy bots etc.

Loosely described, crawlers/spiders/Bots are automated software programmes operated most commonly by search engines, which traverse the web following hyperlinks in web pages and gathering first textual data and then other data to generate indices. The earliest crawlers were very much 'general purpose' by design and were programmed to crawl the web fairly indiscriminately paying little attention to the quality or content of pages with more emphasis on quantity. The goal here, simply being to collect as many pages as possible. As the web, relatively speaking, was a much smaller network then, they were robust enough to discover new web pages and index them concurrently.

Brian Pinkerton (he gets mentioned a lot in this text) notes that, in the early days of WebCrawler, when not enough new pages or URL's were discovered following a crawl, he would download entire newsgroups to "suck out" the links which people placed in their postings so that he could feed them back to the crawler module. As the web has grown, many problems have been encountered by crawlers, including scalability, fault-tolerance and bandwidth restriction. The rapid growth of the web has defeated the capabilities of systems which were not designed to scale-up or to handle the load encountered. Trying to operate a suite of programmes concurrently at such levels without crashing the system became impossible.

Today's crawlers (following research which has only taken place over the past few years as the web has grown) have been modified completely in the relatively short space of time since the early Bots and although fundamentally, they still use the same basic technology, they are now programmed to take a much more 'elegant' approach using proprietary, scalable systems.



General anatomy of a crawler-based search engine

Although crawling is actually very much a rapid process, conceptually, a crawler is doing just the same thing as a surfer.

In much the same way as your browser i.e. Internet Explorer, sends HTTP requests (hypertext transfer protocol), the most common protocol on the web, to retrieve web pages to download and show them on your computer monitor, the crawler does similar, but downloads the data to a client (a computer programme creating a repository/database interacting with other components). First the crawler retrieves the URL and then connects to the remote server where the page is being hosted.

It then issues a request (GET) to retrieve the page and its textual content, it then scans the links the page contains to place in a queue for further crawling. Because a crawler works on 'autopilot' and only downloads textual data and not images or other file types (in the main) it is able to jump from one page to the next via the links it has scanned at very rapid speeds.

The crawler starts with either an single URL or a seed set of pages e.g. possibly pages indexed at Yahoo! as already mentioned, which it then downloads, extracts the hyperlinks and then crawls the pages which are pointed to by those links. Once a crawler hits a page with no other links to follow, it backs up a level and jumps to links it may have missed earlier, or those links which have been programmed in the queue for future crawling. The process is repeated from web-server to web -server until there are no more pages to download or some resources i.e. time, network bandwidth or a given metric has been reached or exhausted.

Crawlers use traditional graph algorithms to traverse the web. The graph is composed of what are known as nodes and edges.

The word 'crawler' is almost always used in the singular, however, most search engines actually have a number of crawlers with a 'fleet' of agents carrying out the work on a massive scale. For instance, Google, as a new generation search engine, started with four crawlers, each keeping open about three hundred connections. At peak speeds they downloaded the information from over 100 pages per second. Google (at the time of writing) now relies on 3,000 PC's running Linux, with more than 90 terabytes of disk storage. They add 30 new machines per day to their server farm just to keep up with growth. Inktomi was the forerunner of using workstation computers to achieve what only super computers had previously managed and started with a cluster of hundreds of Sun Sparc workstations crawling over 10 million pages per day. It's all a major shift from when Brian Pinkerton's innovative WebCrawler ran on a single 486 machine with 800MB of disk and 128MB memory storing pages from only six thousand web sites!

Crawlers use traditional graph algorithms to traverse the web. The graph is composed of what are known as nodes and edges. The nodes (as in a point in a computer network) are the URL's and the edges are the links embedded in the pages. 'Out-edges' are forward links from your web pages which point to other pages and 'in-edges' are back links, those links which point back to your pages from somewhere else (when we come to the connectivity graph/server later in this section, this is described as in-degree and out-degree). By viewing the web in this way, the web graph can be explored mathematically for crawling purposes by using algorithms to produce either a 'breadth first' or a 'depth first' traversal.

Breadth-first crawling means retrieving all pages around the starting point of the crawl before following links further away from the start. This is the most common way that Spiders follow links. Alternately, a depth-first crawl can be used to follow all the links from the first link on the starting page, then the first link on the second page and so forth. Once the first link on each page has been visited it then moves onto the second link and then each subsequent link in order.

The preferred method of a (usually modified) breadth-first crawl provides the benefit of helping to reduce the load on web properties (servers) which is distributed more quickly and helps to avoid 'tying up' domain hosts so that no single web server needs to respond to rapid requests for downloads on a time-after-time constant. A depth-first crawl is easier to programme than breadth-first, but may also result in adding less important pages and missing 'fresher' additions to the web because of the scope.

A research project carried out in 1999/2000 by Compaq Systems Research Centre in conjunction with Alta Vista, called The Mercator Project (Mercator was the Flemish cartographer who was the innovator of a map which gave an accurate ratio of latitude to longitude and also the person who introduced the term 'atlas' for a collection of maps.), revealed that, breadth-first searching does yield higher quality pages. [Najork, Wiener] The Mercator crawler was transferred to Alta Vista at the close of the project as Alta Vista's G3 search engine.

As to how deep into a web site a crawler should go is an issue in itself. A lot depends on the composition of the actual web sites encountered during the crawl and what pages the search engine already happens to know about in its database. In many cases the more important information is near the starting point and the lower the pages in a web site hierarchy the less important they are deemed. There is a logic to this, in that, it would make sense from a design point of view to ensure the more important information a surfer may be looking for, the closer it is from the starting point. You only need to go online and surf for a short while to discover that there are no real rules or standards from a design point of view on the web, so some sites may have lists of links closer to the starting point, with the important information deeper down in the structure. Search engines generally prefer to go for the shorter URL's on each server visited, using the theory that URL's with shorter path components are more likely to be the more general (or useful) pages. This means that, as a very basic example:

http://www.mycompany.com/blue_widgets.html

would likely be deemed more important than something like:

http://www.mycompany.com/products/webcatalog/widgets/blue/spec~series9.html

or something even longer, which is much deeper in a web site hierarchy. Spiders can be programmed/limited in the number of directories they will 'drill down' to in a site by the number of slashes in the URL. Ten directories (slashes) is about the maximum depth count, but in the main, evidence suggests that the third level (directory/slash) is the average depth.

Important pages which are much deeper in a site may have to be submitted to search engines directly by the web site owner. With the constant evolution of the web and related technologies like ASP, PHP, and Cold Fusion, it's very much the case that, many important pages are now 'hidden' deep inside online databases (see section on problem pages).

Initially, crawlers basically collected text to be placed in a repository for indexing with a separate collection point for links (URL's). Now, crawlers collect text, metadata, HTML components, alternative file formats and URL's for analysis and further crawling. One of the other things they need to do is what's known as a DNS lookup (this is the identifier for the domain name server which hosts a particular web site), but these days, by using advanced caching technology, this can be done as a secondary process (cache is vital in web technology).

At any time each of the connections the crawler has open it can be in a different state i.e. connecting to host, sending request, receiving response or doing the DNS lookup. If you check the log files of your web site you'll frequently see names like scooter or googlebot (respectively the names of the spiders for Alta Vista and Google) which means that some (possibly even all) of your pages have been crawled and any relevant information has been extracted (see the 'No Robots' section of the guide for spider names).

'Tying up' web properties was referred to earlier on in this section. This is very much a problem which search engines have had to address in order to be more 'polite' with their online operations. Because crawling is an iterative process carried out at great speed (a series of rapid fire requests) by downloading information from millions of web pages every day, it needs to be moderated in some way as it can present problems to both the search engine and the owners of the sites they visit. In the first instance: because search engines use many agents based on different machines this allows them to download pages in parallel (concurrent threads, or simultaneous program paths to process pages in parallel). This has created yet another overhead for them, in that, the crawlers need to communicate with each other in order to eliminate the possibility of the same pages being visited on multiple occasions. Not only would this cause duplication, it also gives rise to the second instance: consuming resources belonging to the organisations they visit i.e. 'bandwidth hogging' which may result in the casual surfer being denied access to a site because search engine robots are running rampant all over it.

> With the constant evolution of the web and related technologies like ASP, PHP, and Cold Fusion, it's very much the case that, many important pages are now 'hidden' deep inside online databases.

Brian Pinkerton notes how, in the early days, WebCrawler brought a number of servers to a standstill by getting caught in a loop and downloading the same information thousands of times. While Sergey Brin and Larry Page of Google recall how they once tried to crawl an online game which resulted in a lot of garbled messages on the monitors of those trying to play the game. Crawling experiments can only be carried out live, so as well as being able to encourage search engine crawlers, you also need to be able to keep them off your site to protect certain areas, or information which you would prefer not to be indexed. The robots exclusion protocol (explained in this guide) does give some small amount of protection from this.

As has already been mentioned, search engines need to keep the database as 'fresh' (up to date) as possible in order to compete on a commercial level. This, therefore, means that the crawler needs to split its resources by crawling both new pages and checking to see if previously crawled pages have changed simultaneously. To put this into perspective, a study of the size of the web by computer scientists in 1999 put it at 800 million pages. At that time, it was also estimated that, to check those pages just once a week, the crawler would have to download 1300 pages per second [Brandman, Cho et al]. In January 2000 Inktomi completed a study of the web and put it at over 1 billion pages. In December 2001, Google announced that it had a reachable 3 billion documents (newsgroups included)

Sergey Brin and Larry Page of Google recall how they once tried to crawl an online game which resulted in a lot of garbled messages on the monitors of those trying to play the game.

This not only helps to illustrate the growth and scope of information on the web, but also to show how difficult it is to maintain a system which will provide the end user with the 'freshest' results. The freshest results, of course, being the most 'important' and up to date pages on the web, insofar as a particular search engine has crawled it. From a search engine optimisers point of view, it's your job to make sure that crawlers find your pages and that you make sure that they remain 'fresh' and 'important'.

Because web server software programmes (e.g. Apache – the web's most popular server) respond to a request for a page by a crawler in exactly the same way as it replies to a request from a browser, this makes it a slightly primitive interaction. A crawler can acquire a lot more information in a lot less time than a surfer.

Again, this causes a number of problems, including the fact that, as we already know, a crawler does not know exactly when to revisit web pages because, typically, it has no idea when those web pages have changed. So if a web page has not changed since the last visit, the whole process of requesting and downloading it is a waste of time (and more importantly, bandwidth which could have been reserved for surfers). If a server only sent the crawler details of pages which had 'known' changes, then this would be a much better use of resources.

Estimates indicate that the major commercial search engines index only 6-12% of pages available on the web. It's obvious from the above that, search engines (which also have bandwidth constraints) have their own bandwidth reduced purely by having to crawl pages which have not changed since the last crawl. If it were the case that they only crawled pages which had 'known' changes they would then be able to crawl and index a much larger percentage of the web.

Another problem that search engines encounter which only adds to the many problems they have, are 'mirror sites'. These are duplicate web sites or duplicate pages on the web. Studies carried out between 1996 and 1998 discovered that up to almost one third of the web could well consist of duplicate pages [Bharat, Broder].

There are a number of reasons why pages/sites are duplicated on the web. Some for technical reasons i.e. providing faster access or creating back-ups on different servers in case one should go down. Technical manuals and tutorials for software and programming languages are literally 'cut and pasted' into web pages and uploaded at the encouragement of the creators or developers (Java FAQ's, Linux Manuals etc. etc.) There are also millions of sites which belong to re-sellers or affiliates who use mirror sites/pages to promote a third party product to earn commissions.

There are millions of pages sharing information e.g. in the scientific community where certain key papers are posted on many servers on the web. And let's not forget the Spam! Millions of pages are either duplicates or near duplicates attempting to dominate search engine rankings for specific keywords or phrases. Not only does this mean that search engine databases get full of duplicate material which is only taking up space and impeding both the progress (in terms of scope) and bandwidth allocated to the crawler: it also means duplicate results following a query. The end user of a search engine is deeply unsatisfied if the same information is discovered after clicking through the top ten results.

To combat this, search engines have developed sophisticated techniques (algorithms) to filter out duplicate or near duplicate documents and then limit the number of pages returned following a query to only one (usually with an option of 'more pages from this site' or 'similar pages').

There are legitimate reasons for having duplicate material i.e. two versions of a site designed for different monitor resolutions, but even in this type of instance you would be better off looking at no robots .txt files on one of the mirrors to avoid potentially being penalised for spamming (the Spam is filtered out - see section on Spam).

Another group of leading experts in the field of crawling the web published a paper which took a comprehensive overview of the problems search engines encounter [Brandman, Cho et al - 2000]. The paper was innovative and suggests that many of the current problems could be alleviated if crawlers and servers used a more appropriate protocol. One way this could be done is by the web server providing metadata (data about the data the server was hosting) in advance. If the server kept an independent list of all URL's and their metadata (last modified, description etc.) specifically for crawlers, they could then use the information prior to downloading to identify only pages which had changed since the last crawl. They would then only request those pages. This also provides another benefit, in that, it would become easier to estimate the frequency of changes to pages as well as being able to calculate the amount of bandwidth required to refresh those pages before crawling.

Using a newer web technology/language such as XML could make this possible. But this already has the initial drawback of providing web-wide Spamming opportunities. XML is too complicated and beyond the scope of this guide for a detailed explanation. In short, XML is the Extensible Markup Language. It's designed to improve the functionality of the Web by providing more flexible and adaptable information identification.

It's called extensible because it's not a fixed format like HTML. XML is actually a 'metalanguage', a language for describing other languages which lets you design your own customised markup languages for limitless different types of documents.

Although I have broken down the anatomy of a search engine into five distinct components/modules, for the purpose of making it easier to understand the process, it has to be noted that this is not a linear process. A multitude of operations can be taking place at any one time between the "suites of computer programmes" as I referred to them earlier. What I'd like to do now, is a kind of 'segue' into the search engine database and take a look at what's happening there while the crawler is 'doing its stuff".

THE REPOSITORY/DATABASE MODULE

Once the search engine has been through at least one crawling cycle, primarily, the database itself is the focus for all future crawling decisions i.e. it becomes crawler-control. From what you will have gathered so far, clearly there has to be an ordering metric for crawling 'important' pages, with billions of URL's beginning to amass in the crawler control queue.

Starting a crawl with a number of URL's which may even share a topic, or a series of topics (think of the Yahoo! example given earlier) and following the aggregated links, soon leads to completely off topic pages and thousands upon thousands of links leading to thousands upon thousands of pages without cohesion.

A general purpose crawler (as opposed to a focused crawler covered later), such as Google, needs to be able to provide relevant results for over 150 million queries, covering hundreds of thousands of topics, every day of the week [Silverstein 2001]. But with a database full of text, html components, subject headings from directories, data supplied by marketing partners, alternative file formats, and literally billions of URL's etc. (let's not forget there's also tons of Spam in there): how can they sort and rank all of this data into some sort of order to be able to provide the most relevant results? And how can they decide which links to follow next, which links to revisit and which ones to dump?

> Search engines have developed sophisticated techniques (algorithms) to filter out duplicate or near duplicate documents

Once again, Brian Pinkerton's innovative early work helps here to give an indication of how a simple web crawler can evolve from being a basic page gathering tool to a large database-based system. And also how current search engines have used this experience to further evolve the process and the technology.

At first, WebCrawler was able to download pages from the web, retrieve the links from those pages for further crawling and then feed the full text of the page into the indexer concurrently.

Quickly, with the proliferation of more and more pages on the web, the process had to be separated into collaborative functions starting with the indexing becoming a 'batch' process which ran nightly.

Fundamentally, the basic link analysis being carried out by WebCrawler would eventually develop into one of the single most important factors for determining 'important' ('hot pages') by all crawler based search engines.

Even in the early days, WebCrawler's database contained a link table which kept information on relationships between documents. A link from document A to document B resulted in an {A,B} pair in the link table. At that time, this data was not used to influence crawling, but added the novelty of being able to present the 'WebCrawler Top 25' (the 25 most linked to sites on the web). As for the crawling policy, that still worked on a first in first out basis (although more emphasis was placed on an URL with the string 'index.html' or 'index.htm' if it were known to exist). By the time of the second crawler, it was the 'back-links' which had become the best way to identify 'important' pages in the database for future crawling i.e. page P is more important to crawl than page Q if it has more links pointing to it from pages not on the same server.

Fundamentally, the basic link analysis being carried out by WebCrawler would eventually develop into one of the single most important factors for determining 'important' ('hot pages') by all crawler based search engines.

Search engine optimisers simply call it the 'link popularity factor', but search engines use different algorithms based on linkage. For search engines it's about 'PageRank', 'hubs and authorities', 'citation and co-citation' and 'neighbourhood graphs'.

There are other 'heuristics' which search engines use to determine 'important' pages, for instance, crawler control may also use feedback from the query engine to identify 'hot' pages (links which are most frequently clicked on following specific keyword searches), or pay more attention to pages in a certain domain i.e. .com or .gov. However, you can be certain that, connectivity and link anchor text, provide search engines with the most significant information for identifying 'hot' pages.

Purely for information, you may be interested to know that, a study of the web's connectivity map carried out by Andrei Broder and colleagues suggests that the link structure of the web can be visualised as looking like a "bow-tie". His research reveals that about 28% of the web pages constitute a strongly connected core (the centre of the bow tie). About 22% form one of the tie's loops: those are pages which can be reached from the core but not vice versa. The other loop consists of 22% of the pages which can reach the core but cannot be reached from it. (The remaining nodes/links neither reach the core nor can be reached from it).

Without going too far off topic (for want of a better phrase as you will see), I should mention that there is a much more detailed version of the results of the 'cyberspace mapping' experiment. For his part, Andrei Broder won the 'Scientific paper of the year' award.

So, back to the repository/database with its very large collection of data objects. Each web page retrieved by the crawler is compressed and then stored in the repository with a unique ID associated with the URL, and a note is taken of the length of each page (it's important to put your most relevant information high in a web page as, Google, for instance only downloads the first 110k of a page, Alta Vista only downloads the first 100k so be careful with long/heavy pages). All URL's are resolved from the relative path to the absolute path and then sent to the URL server to be placed in the queue of pages to be fetched. It's here in the link table where the ordering metrics for future crawling need to be determined. There could be a number indexes built on the content of the pages. The link index and the text index are the main indexes, but utility indexes can be built on top for, say, PageRank in the case of Google or different media types images etc. As I've already mentioned (and need to do so again) the entire process of crawling the web, downloading pages and ranking and returning documents to user queries is enormously complex.

So, here I'll give a simple outline of the three basic methods which search engines can use for evaluating the 'importance' of web pages for both crawling and re-crawling.

Textual Similarity

This is where analysis of user queries is important. The words which are used in the query are matched against pages in the database containing the same words. The similarity to the query is gauged by the number of times the word appears in the document and where in the document it appears. The pages which are returned most frequently to a specific query using this metric are those deemed to be the most relevant and therefore have significant importance for further crawling.

Page Popularity

A popular page can be defined by the number of other pages which point back to it. This can also be referred to as 'citation count' i.e. where one web page views another page to be important by referring to it (pointing a link to it). The more links (citations) a page has pointing to it, the more important (popular) or of general interest it appears to be. This use of 'bibliometrics' on the web is derived from the way that published papers are evaluated by citation (covered later).

Page Location

This type of metric relies solely on the location of the page on the web and not to its contents. Specific domains such as .com, or .co.uk may have a higher degree of importance than others.

Certain URL's which contain the string "home" or another page identifier in it may be deemed as likely to be more useful. It's a known fact that, Google (amongst the many other factors taken into account including PageRank) prefers .gov and .edu pages. [see interview with Craig Silverstein]

Let me just take a real example of scoring here. For two years at Brian Pinkerton's WebCrawler it worked like this (no parallel assumptions should be made here as this pre-dates this text by a long time and it pertains only to WebCrawler – this proves as an example of a genuine crawling algorithm only).

Each document is awarded an unbounded score that is the sum of:

15 If the document has ever been manually submitted for indexing.
5 For each time the document has been manually submitted
7 If any other document in the database links to the document
3 For each inbound link
7 If the URL ends in / or
5 if it ends in .html
1 For each byte by which the path is shorter than the maximum (255)
20 If the hostname is a host name and not an IP address
5 If the host name starts with www
5 If the host name's protocol is http
5 If the host's URL scheme is https
5 If the URL is on the default port for http (80)
1 For each name by which the name is shorter than the maximum

[Pinkerton – WebCrawler Thesis 2001]

The above should give an indication of the kind of logic used by crawler control to determine which pages in the database are already 'hot' and those which are likely to be 'hot' for crawling. The repository keeps feeding links into crawler control and forwards the full text from the pages, as well as the link anchor text to the indexer. Because the repository will frequently contain numerous obsolete pages i.e. pages which have been removed from the web after a crawl has been completed, there has to be a mechanism in the system for it to be able to detect and remove 'dud' pages.

THE INDEXER/LINK ANALYSIS MODULE

As you are now aware, in order to make it easier to grasp the concept of how search engines work, I thought it would be easier to break it down into a series of components. I have to point out yet again though: it's not a linear process (even if the algebra is!). Many things are happening at the same time and certain components are more closely linked than others.

> It's important to put your most relevant information high in a web page as, Google, for instance only downloads the first 110k of a page, Alta Vista only downloads the first 100k so be careful with long/heavy pages.

As already noted, there has been much work in the field of information retrieval (IR) systems. Statistical approaches have been widely applied because of the poor fit of text to data models based on formal logics e.g. relational databases.

So rather than requiring that users will be able to anticipate the exact words and combinations of words which may appear in documents of interest, statistical

IR lets users simply enter a string of words which are likely to appear in a document. The system then takes into account the frequency of these words in a collection of text, and in individual documents, to determine which words are likely to be the best clues of relevance. A score is computed for each document based on the words it contains and the highest scoring documents are retrieved.

Three retrieval models have gained the most popularity: Boolean Model; Probabilistic model; Vector Space Model. Of particular relevance to search engines happens to be the work carried out in the field of automatic text retrieval and indexing. Pre-eminent in the field is the late Gerard Salton who died in 1995. Of German descent, Salton was Professor of Computer Science at Cornell University, Ithaca, N.Y. He was interested in natural-language processing, especially information retrieval, and began the SMART information retrieval system in the 1960's (allegedly, SMART is known as "Salton's Magical Automatic Retriever of Text"). Professor Salton's work is referred to (cited) in just about every recent research paper on the subject of information retrieval.

Salton developed one of the most influential models for automatically retrieving documents in 1975. Known as the Vector Space Model, it was designed to specify which documents should be returned for a given query and how those results should be ranked relative to each other in the results list. This model is still very much fundamental to the index and retrieval systems of full text search engines. In his own words, here's how he describes the model:

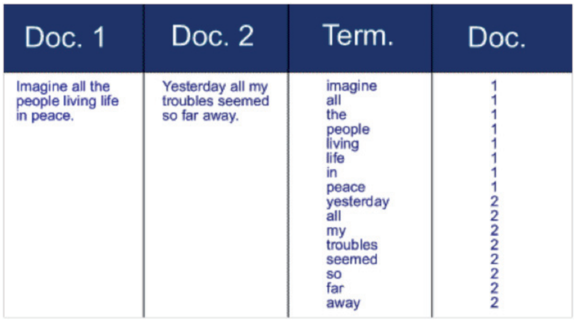
"In a document retrieval, or other pattern matching environment where stored entities (documents) are compared with each other, or with incoming patterns (search requests), it appears the best indexing (property) space is one where each entity lies as far away from the others as possible; that is, retrieval performance correlates inversely with space density. This result is used to choose an optimum indexing vocabulary for a collection of documents."

There! Simple enough I would have thought… no? O K, joking apart, I'll try to give a simple (and I do mean simple) explanation of how the full text index is inverted and then converted to what are known as 'vectors' (vector: a quantity possessing both magnitude and direction).

First of all, remember, that, the crawler module has now forwarded all of the 'raw data' to the repository and parsed the HTML (extracted the words). The repository has given each item of data its own identifier and details of its location i.e. URL. The information is then forwarded across the search engine's distributed system.

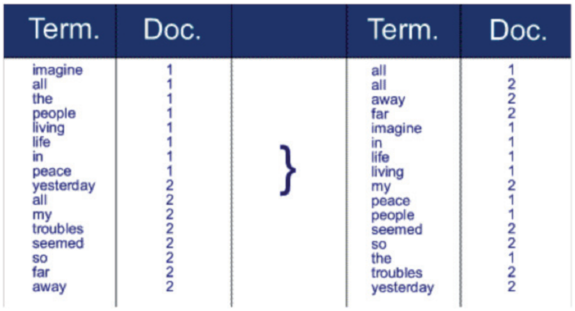
The words/terms are saved with the associated document (Doc) ID in which it appeared. Here's a very simple example using two Doc's and the text they contain.

Recall Index Construction.



Doc. 1	Doc. 2	Term.	Doc.
Imagine all the people living life in peace.	Yesterday all my troubles seemed so far away.	imagine	1
		all	1
		the	1
		people	1
		living	1
		life	1
		in	1
		peace	1
		yesterday	2
		all	2
		my	2
		troubles	2
		seemed	2
		so	2
		far	2
		away	2

After all of the documents have been parsed the inverted file is sorted by terms:



Term.	Doc.		Term.	Doc.
imagine	1		all	1
all	1		all	2
the	1		away	2
people	1		far	2
living	1		imagine	1
life	1		in	1
in	1		life	1
peace	1	}	living	1
yesterday	2		my	2
all	2		peace	1
my	2		people	1
troubles	2		seemed	2
seemed	2		so	2
so	2		the	1
far	2		troubles	2
away	2		yesterday	2

In my example this looks fairly simple at the start of the process, but the postings (as they are known in information retrieval terms) to the index go in one Doc at a time. Again, with millions of Doc's, you can imagine the amount of processing power required to turn this into the massive 'term wise view' which is simplified above, first by term and then by Doc within each term.

This data is the core component when it comes to retrieval following a user query, by both effectiveness and efficiency. Effectiveness measures the accuracy of the result in two forms: precision and recall. Precision is defined as the fraction of relevant documents retrieved to the total number of documents retrieved (covered more specifically later in this section). Recall (as shown above) is defined as the fraction of relevant documents to the total number of documents in the collection.

Efficiency measures how fast the results are returned (note how Google will always give a precise time for effectiveness following a search i.e. Results 1 - 10 of about 34,900. Search took 0.10 seconds).

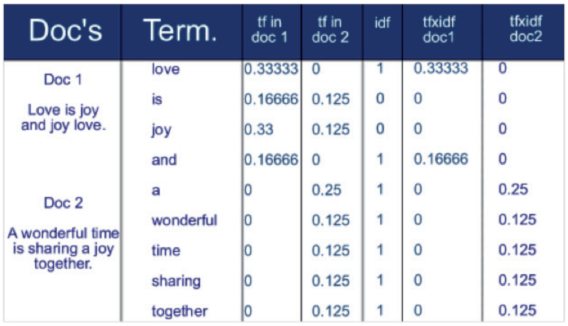
Each search engine creates its own custom dictionary (or lexicon as it is – remember that many web pages are not written in English) which has to include every new 'term' discovered after a crawl (think about the way that, when using a word processor like Microsoft Word, you frequently get the option to add a word to your own custom dictionary i.e. something which does not occur in the standard English dictionary). Once the search engine has its 'big' index, some terms will be more important than others. So, each term deserves its own weight (value). Here, the indexer works out the relative importance of:

0 vs. 1 Occurrence of a term in a Doc.
1 vs. 2 Occurrences of a term in a Doc.
2 vs. 3 Occurrences of a term in a Doc and so forth.

A lot of the weighting factor depends on the term itself i.e. (Andrei Broder gives the example): what tells you more about a doc? Ten occurrences of the word 'haemoglobin' or ten occurrences of the word 'the'? Of course, this is fairly straight forward when you think about it, so more weight is given to a word with more occurrences, but this weight is then increased by the 'rarity' of the term across the whole corpus. The indexer can also give more 'weight' to words which appear in certain places in the Doc. Words which appeared in the title tag <title> are very important. Words which are in <h1> headline tags or those which are in bold <b> on the page may be more relevant. The words which appear in the anchor text of links on HTML pages, or close to them are certainly viewed as very important. Words that appear in <alt> text tags with images are noted as well as words which appear in meta tags (see section on keywords and writing for the web). Taking these textual occurrences into account, I'll take a look at what's hot and what's not for re crawling and remaining in the index later.

To summarise: a full text index is an inverted structure which maps words to lists of documents containing them and the relative importance of the documents. Each search engine also incorporates a thesaurus at this stage to map synonyms.

Once this is achieved the indexer then measures the 'term frequency' (tf) of the word in a Doc to get the 'term density' and then measures the 'inverse document frequency' (idf) which is a calculation of the frequency of terms in a document; the total number of documents; the number of documents which contain the term. With this further calculation, each Doc can now be viewed as a vector of tf x idf values (binary or numeric values corresponding directly or indirectly to the words of the Doc). What you then have is a term weight pair. You could transpose this as: a document has a weighted list of words: a word has a weighted list of documents (a term weight pair).



Doc's	Term.	tf in doc 1	tf in doc 2	idf	tfxidf doc1	tfxidf doc2
Doc 1 Love is joy and joy love.	love	0.33333	0	1	0.33333	0
	is	0.16666	0.125	0	0	0
	joy	0.33	0.125	0	0	0
	and	0.16666	0	1	0.16666	0
Doc 2 A wonderful time is sharing a joy together.	a	0	0.25	1	0	0.25
	wonderful	0	0.125	1	0	0.125
	time	0	0.125	1	0	0.125
	sharing	0	0.125	1	0	0.125
	together	0	0.125	1	0	0.125

Now that the Doc's are vectors with one component for each term, what has been created is a 'vector space' where all of the Doc's live (space in mathematics is a set with structure on it, especially geometric or algebraic structure).I could take up another three pages, at least, attempting to describe this multi-dimensional space, but this would overly complicate the issue when I'm trying to keep it to the basics. It would be much easier to grasp this if it were possible to come up with a good analogy. I've heard many including star charts and street maps (albeit both of those out of context). Mine may be just as lame (and nowhere near as dimensionally diverse as the model itself) but perhaps more in context with the subject matter. Maybe when William Gibson first coined the term 'cyberspace' and because we are used to using the term when it comes to the web, that's about as close as we can get for an analogy.

After all, the web itself is full of computers hosting URL's full of words and each one being a reference point in space, many with a connectivity or relevancy which links them together, and many that do not. This is a data 'space' in which everything has its own coordinate.

But what are the benefits of creating this universe of Doc's which all now have this magnitude? In this way, if Doc 'd' (as an example) is a vector then it's easy to find others like it and also to find vectors near it. Intuitively, you can then determine that, documents which are close together in vector space, talk about the same things. When the term weights are 'normalised' so that longer pages don't get more weight, the retrieval engine can then look for what are known as 'cosine similarities' or the 'vector cosine' (that's correlation to us laymen, by the way). It's very difficult to explain all of this without getting into some of the math here, at this point let me just say that, this means being able to sort Doc's by similarity i.e. Doc's which contain only frequent words like 'the', 'and' etc. or Doc's which have many rare words in common like 'anaemia', or 'haemoglobin'. By doing this a search engine can then create clustering of words or Docs and add various other weighting methods.

However, the main benefit of using term vectors for search engines is that, the query engine can regard a query itself as being a very short Doc. n this way, the query becomes a vector in the same vector space and the query engine can measure each Doc's proximity to it. The Vector Space Model allows the user to query the search engine for 'concepts' rather than a pure 'lexical' search using Boolean logic which most surfers don't understand or may not be aware exists.

Intuitively, you can then determine that, documents which are close together in vector space, talk about the same things.

To try and explain this process more comprehensively, I'm afraid I have to refer to the computation involved. This is not essential reading and very difficult to follow, but I felt it was necessary to include to substantiate some comments I'll be making later in this section of the guide.

Let's disregard Boolean operators and simply assume that a user query is just a list of terms (as the norm at search engines). Each term in the query is then associated with a 'query term weight', let's make this query term weight constantly 1. On the other side, the terms in each document get a 'document term weight'. The weight is the product of a document specific weight and the 'inverse document frequency' (as described above). The latter being defined as 'idf=log(P/p) for instance, where P is the number of Doc's in the database and p being the number of Doc's the term appears in.

Now, the other part of the document weight is computed like this: Let 'tf' be the number of occurrences of the term in the document and 'maxtf' the maximum frequency of any term in the Doc. A preliminary weight is computed according to 'x=(0.5xtf)/(1+maxtf). These weights are then normalised by dividing them by the sum of the squares of all preliminary weights for terms in this Doc. The document specific weights make up a vector length of 1. And then the final document term weight is yielded by multiplying this weight to the 'idf'.

So, for simple queries (no Booleans) the weight of a document is computed by multiplying the term weight to the query term weight for each term in the query and calculating the results. This is what is referred to as the 'vector product' (correlating to the name of the Vector Space Model and also known as the 'scalar product').

There is more to it than this, but those readers who understand the math (in its most simplistic form as I have it) will have registered it already. For those who don't – please don't worry about it, as this a simple example of how it works, and you needn't be too concerned about whether you understand it or not - for the purpose of this guide, it just has to be here for future reference.

Brian Pinkerton used this ("classic Salton approach" as he calls it) Vector Space Model with the first Web-Crawler.[Note that Michael Mauldin of Lycos also makes reference to the same approach as far back as 1994].

Brian explains the process as he used it as follows: 'Following a query, documents in the result set were sorted and ranked on how closely the words in the query matched the words in the document. The more closely they matched, the higher they would rank.

Typically (though not necessarily) a word is more important in one document than another if it occurs more frequently in that first document. This model works well for surfers using long queries, or where there are only a few good document matches for the query. However, it falls down where the query is very small as the vector model doesn't distinguish among the resultant documents very well'. This is something which Larry Page and Sergey Brin made note of in their research papers for Google. Following a search for the query 'Bill Clinton' the top result was a page which simply had a picture and the words 'Bill Clinton sucks'. If there was another page which existed and it was a Whitehouse page with exactly the same composition i.e. a picture and a headline which said 'Bill Clinton, President' – how would the search engine know which was the best page to return? It wouldn't. In isolation, the Vector Space Model fails because of the immense size of the corpus and the use of extremely short queries. This is why, in the second phase of WebCrawler, it moved to a full-blown Boolean query model with 'phrase searching' and proximity boosting.

There have been many variants to attempt to get around the 'rigidity' of the Vector Space Model. For instance, in 1999 a research team in China proposed an extended Vector Space Model to attempt to take into account 'natural language processing' and 'categorisation' [Xhiohui, Hui, Huayu]

There is much talk about 'themed' web sites in SEO circles. I want to cover this in more detail when touching on the term vector database, which (as already mentioned) many people have confused with the Vector Space Model. When talking about themes, most refer to a pair, or sequence of a few words which can vaguely give a characteristic of the page itself. As we now know, with Brian Pinkerton's reference to 'phrase searching' this is not new. Search engines use a clever extension to the ranking algorithm for multi-word queries with no operators. Think about it this way, If a surfer issues a query for something like: 'hotels in new york'. The surfer is obviously looking for a hotel in the city/state of New York. The basic Vector Space Model simply takes the terms independently without any attention to the actual phrase 'New York'. The modified algorithm first weights Doc's with all of the terms more highly than those which have only some of the terms and then weights terms which occur as a phrase more highly than those that do not.

Although I have said that of the three main retrieval models used by search engines, the Vector Space Model is of the greatest interest, you'll note that Brian Pinkerton also mentioned that he eventually had to add a "full-blown Boolean query model with 'phrase searching' and proximity boosting". It has to be said here about Boolean operators that, the purpose of a search engine could be better served if we all understood Boolean logic. Once again, Andrei Broder pointed out that most queries to search engines are nothing more than a simple (and very short) series of text strings. A search engine which employs a retrieval system handling both concept searches (Vector Space Model) and Boolean can satisfy both the casual surfer and the serious searcher.

> I wonder if the great mathematician Al-Khwarizmi [Born 770 Uzbekistan] would ever have expected that his name would be bandied around as much as it is in the 21st century.

Adding simple Boolean operators such as AND, OR, NOT as well as ADJ (for adjacent) which is a critical 'phrase' operator is usually satisfactory. But most search engines, even when they employ Boolean logic, are dominated by 'novice' searchers (the casual surfers) who don't issue well well-formed Boolean queries.

Some search engines also employ 'Porter Stemming' which is an algorithm for 'suffix stripping' named after its developer Professor Martin Porter. The algorithm was originally described in 1980 and it removes plurals and suffixes i.e.

CONNECTED
CONNECTING
CONNECTION
CONNECTIONS

Can all be stemmed back to CONNECT. Using the full Porter model is usually a bit too aggressive for search engines so a modified or restricted version has been used mainly for plurals i.e. CONNECTIONS to CONNECTION.

During the course of the research for this second edition, something occurred to me in relation to the Porter stemming algorithm, that being it appears only to work with English language text. So what about all of the non English pages which are stored in search engine databases?

Using Porter stemming is good for saving space in the index, but surely it must cause problems if a search engine applies it to all languages. After puzzling over this for a while, I had the answer – ask Professor Porter himself! I was very grateful when he qualified this for me:

"Mike, you are quite right that there is no point in applying the Porter stemmer to anything other than text in English. On the other hand, applying it to non-English texts does not usually do great harm, so you can try the blanket approach of passing everything through the Porter stemmer if you know that, say, only 5% of the material is non-English. Obviously there are various approaches. Google does not do any stemming, so singular and plural variants have to be searched for separately. Ideally, you would like a search engine to divide the Web by language. Nor is language identification so hard: there are various ways of doing it, and the one I've used is just counting up small particle words. 'the' 'a' 'at' 'of' implies English; 'le' 'la' 'de' 'dans' French and so on. But the problem is how to make use of this information. You can apply different stemmers to different languages, but then the user has to declare which language the query is in, so that the same stemming process can be applied to the query. But many queries are for proper names ('Maradona', 'The Beatles', 'Charles Dickens'), and not therefore language specific. Besides, there are so many languages in use on the Web that providing linguistic normalisation tools for all of them is not really practicable".

If the Vector Space Model was used on its own, it's a simple enough thing for a search engine optimiser to simply add a specific word (sometimes in invisible text on the page) a few hundred times and up that page flies to the top of the results.

Later in this section I'll be covering bibliometrics, but it merits a reference here when I mention 'stop words' (most search engines remove common words or 'stop words' (particles) like 'and', 'of', 'the', from user queries). The connection of 'stop words' to bibliometrics is this: The most powerful, wide ranging law of bibliometrics is Zipf's Law which is named after Harvard Linguistic Professor George Kingsley Zipf. Essentially this law predicts the phenomenon that as we write, we use familiar words with high frequency. Zipf said his law is based on the main predictor of human behaviour: striving to minimize effort. Therefore, Zipf's work applies to almost any field where human production is involved. This means we also have a constrained relationship between rank and frequency in natural language. And this law is confirmed by the existence of 'stop lists' at search engines.

So, to sum up this far in the indexing phase: The construction of the full text index covered above uses algorithms to analyse, weight and sort Doc's by certain 'on the page' criteria (text, HTML tags, meta tags etc.) and places great importance on the use of the Vector Space Model and Boolean operators.

THE RETRIEVAL/RANKING MODULE

Other algorithms for 'off the page' criteria now pay a much more important role in the way that pages are ranked. Before I explain why 'off the page' criteria, or heuristics, are so important, I think it may be a good idea, having used the word algorithm so many times, to try and explain what an algorithm is.

I wonder if the great mathematician Al-Khwarizmi [Born 770 Uzbekistan] would ever have expected that his name would be bandied around as much as it is in the 21st century. As it's from his name that the term algorithm is derived. As I noted earlier about the crawler module being referred to in the singular, the same happens here with algorithm. But as you can see, there are many algorithms used by search engines. Just the use of the word algorithm can strike awe into the uninitiated. For sure, an algorithm developed by a search engine scientist reads like Greek to a non mathematician like myself (and perhaps you) but when it is explained in its simplest form, it's not too hard to grasp at all.

Algorithms are the fundamental basis for the performance of computer programmes. An algorithm is a set of instructions to automatically complete a task. In fact the word algorithm could be used to describe any automated task or list of instructions. Let's see it for what it is. We all use labour saving devices to aid us in what can be simply intensive and boring. A washing machine can now be programmed to wash, spin and dry to save us the tedious bother.

Yes - it uses an algorithm to perform a routine set of tasks. How can I even more easily describe an algorithm? Here's an algorithm I frequently use myself:

Go into the lounge.
Find a small black plastic object with buttons which can be held in the hand.

Point it at the TV and press button number six for football game.
Go to the fridge and take out a cold beer.
Sit down in armchair and remove opening device from can.
Place can to lips and drink.

Of course, you could do the same thing by getting the beer first and then putting its contents into a glass before you go to the lounge and sit down to press button number six on the hand held device. Which is the best algorithm? Well that just depends on the person and the circumstances. Do you prefer to drink out of a can or a glass; And would you put the TV on first – or get the beer first?

The good thing about an algorithm is that there is no complaint. Things we are expected to do as humans which could take forever through lack of interest and tedium, a computer will do until told not to. I have many trees in my back garden (Yard). Each Autumn (Fall) thousands of leaves of different types drop onto the lawn. I assume there are thousands, but I don't really count them. Some blow over the fence without touching the lawn, so it could be more or less. With an algorithm designed to monitor the environment, it would be a simple and precise process to find out if I'm right or not.

Ok, let's go back to the 'on the page' - 'off the page' subject. If the Vector Space Model, even combined with Boolean logic operators works as well as it does: why do we need to look at algorithms designed around 'off the page' criteria? There are a number of reasons which will be covered, but one of the most important to commercial search engines is the problem of Spam. There is a section in the guide which covers Spam, but here, let's look at the immediate problem where a purely text based sort and rank algorithm fails on the web. When I covered the section on 'weighting terms' I explained how certain terms get greater weight. What this means is, it's not too difficult to manipulate the results by web page authors adding documents to the search engine index with an artificial density. So, if the Vector Space Model was used on its own, it's a simple enough thing for a search engine optimiser to simply add a specific word (sometimes in invisible text on the page) a few hundred times and up that page flies to the top of the results.

For example, if a web page author added the word 'Viagra' a few hundred times to a page, then inevitably that page, under the Vector Space Model, would come to the top of the results when a user keys that query into the search box. Of course, search engines have been wise to this for a long time (note my interview with Brian Pinkerton) and employ a number of filters to 'penalise' pages attempting this form of Spamming.

Alternatively, it's also possible that, the Vector Space Model may not return the most relevant pages on its own following a query, because the keyword density, by default, may be higher on other less important/relevant pages – or it may not appear at all. Let me explain that last bit. Some years ago, before 'off the page' criteria added another important degree of relevancy to search results, you could go to Alta Vista and key in the words 'search engine'. At that time, Alta Vista was the number one search engine online. So you might expect that, following this keyword search – Alta Vista would be number one in the results? Wrong. The Alta Vista home page did not contain the words 'search engine' either on the visible page or in the hidden tags. Here's another one, keying in the words 'Bill Gates', you would expect that the Microsoft home page would be at the top of the results as Bill Gates owns the company. Wrong. The Microsoft home page did not contain the words 'Bill Gates' either.

Go to Google now (as the world's most popular search engine) and key in the words 'search engine' and who comes in at number one? Google. But wait... look at the Google home page: It still doesn't contain the words search engine! So, try this. In the search box at Google, key: link:www.google.com and when the results are returned you'll see in the blue bar at the top of the results page: Results 1-10 of about 350,000 [27/01/2002]. With that many links pointing back to them in their own database and using their own technology for retrieval and ranking, of course they make it to number one. And just before you go to check – yes Bill Gates' own home page at Microsoft comes in at number one (his name is only mentioned once on the page – but there are 4,990 links in the Google database pointing back).

Coupling measures the relationship between source documents, co-citation measures the relations between cited documents.

The example I've used above gives a clue to the importance of 'off the page' criteria used by search engines.

The subject of back-links has already been touched on when I covered the crawler and how the number of back-links could help to identify 'hot' or popular pages to crawl and re-crawl. But this rudimentary data does not reveal a great deal in terms of relevancy. So here, once again, conventional techniques used in information retrieval are applied.

Bibliometrics is a word used in information science to describe the coupling and co citation of documents. I'll try to expand on this to help to explain what current link analysis is based on. Classic bibliometrics result from the idea that information has patterns that can be analysed by counting and analysing citations, finding relationships between these references based on frequency, and using other statistical formulas to establish 'coupling'. As far back as the late 60's early 70's information retrieval software to detect patterns and establish relationships between electronic publications had been designed. [Price 1968] [Schiminovich 1971]

> The connectivity pattern of linkages on the web and the content in the link itself (anchor text) can imply certain information about the importance of a page.

And the idea of bibliometric coupling was developed further by Markova and Small by noting that, if two references are cited together, in a latter literature, the two references themselves are related. The greater the number of times they are cited together, the greater their co citation strength. This further development shows that the difference between bibliometric coupling and Co-citation is, while coupling measures the relationship between source documents, co-citation measures the relations between cited documents. This therefore suggests that an author purposefully chose to relate two articles together and not just show an association or common 'link' between them.

Conventionally, co-citation analysis has been used as a tool to identify a core set of articles, authors, or journals of particular fields of study. This type of analysis is now used in a broad range of disciplines. In fact, citation counting has even been used to speculate the future winners of the Nobel Prize. The way it has been used to provide a kind of mapping of intellectual structure by the topical relatedness of authors, journals or articles, provides a fundamental basis in attempting to view the hyperlinked structure of the web. But in the heterogeneous world of the web it's not an easily applied transfer of thought. Conventional co-citation analysis follows a consistent sequence of steps [McCain 1990]:

Selection of the core items for the study
Retrieval of co-citation frequency information for the core set
Compilation of the raw co-citation frequency matrix
Correlation analysis to convert the raw frequencies into correlation coefficients
Multivariate analysis of the correlation matrix, using principle components analysis, cluster analysis or multidimensional scaling techniques

Interpretation of the resulting 'map' and validation

Once again, I have to point out that I'm trying to keep things as simple as possible, for what is a very complex subject. But there's no other way of describing the 'steps' than the way they have already been defined. In just the same way as the Vector Space Model which I tried to simplify earlier in this text, none of this can be made *so* simple. Hopefully though, you can keep up with the explanation of processes/algorithms, without me having to essentially go too deeply into the math.

By using this methodology search engines can attempt to identify the intellectual structure and 'topology' of the web.

However, there are many problems, as said, in scaling the methods used in co-citation analysis to deal with hundreds and hundreds of millions of documents with billions of citations. The 'buzz' about 'link popularity' as it's known within search engine optimisation circles is fairly new because Google has been so 'visible' about it. Yet this type of experimental research was actually carried out as early as the development of the second phase of WebCrawler and also with Inktomi in preliminary studies at Berkeley.[Larson 1995]

Just as much attention as has been given to text retrieval and indexing is now being given to the structure and linkage of the web. Web connectivity and its 'topology' provide many clues to search engines as to the importance and the content of a given web page.

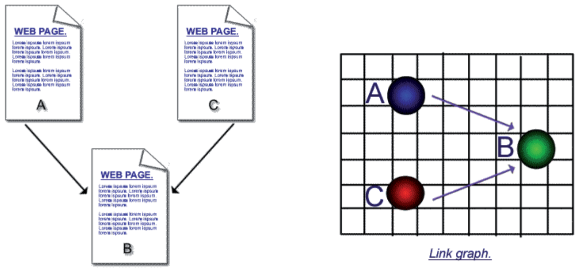
However, the links connecting web pages together, in principle, are equivalent. The web itself holds no preference for one link over another. But the connectivity pattern of linkages on the web and the content in the link itself (anchor text) can imply certain information about the importance of a page.

Some links on web pages are simply navigational aids to 'browse' a site. Other links may provide access to other pages which augment the content of the page containing them. Andrei Broder pointed out that, a web page author is likely to create a link from one page to another because of its relevance or importance: "You know, what's very interesting about the web is the hyperlink environment which carries a lot of information. It tells you: 'I think this page is good' – because most people usually list good resources. Very few people would say: 'Those are the worst pages I've ever seen' and put links to them on their own pages!"

High quality pages with good, clear and concise information are more likely to have many links pointing to them. Whereas low quality pages will have fewer links or none at all. Hyperlink analysis can significantly improve the relevance of search results. All of the major search engines now employ some type of link analysis algorithms.

Using the citation/co-citation principle as used in conventional bibliometrics, hyperlink analysis algorithms can make either one or both of these basic assumptions:

A hyperlink from page 'a' to page 'b' is a recommendation of page b by the author of page a and creates a 'directed edge' in the link graph {A,B}



Set of linked pages.

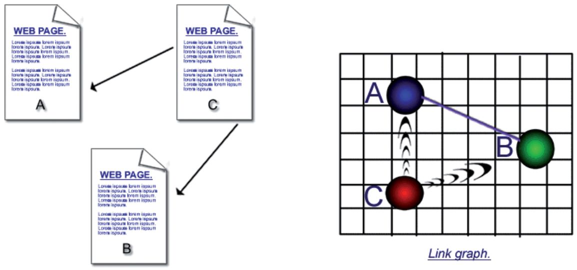
Link graph.

Web pages linked together are nodes in the web graph.

When web page A links to web page B this is a 'directed edge'.

If pages 'a' and page 'b are connected by a hyperlink, then they may be on the same topic.

Some algorithms also use an undirected co-citation graph. A and B are connected by an undirected edge, if and only if there is a third page C which links both to A and B.



Link graph.

Set of linked pages.

If page C links to both A and B, then A and B are connectec by an undirected edge in the graph and are viewed as beir co-cited by C.

Hyperlink analysis provides search engines with much vital information for both crawling and ranking purposes. This information is also useful for discovering the geographic scope of a web page and finding 'mirrored' hosts , duplicate pages etc. In my interview, Andrei Broder explained that, connectivity-based ranking algorithms can fall into two main classes:

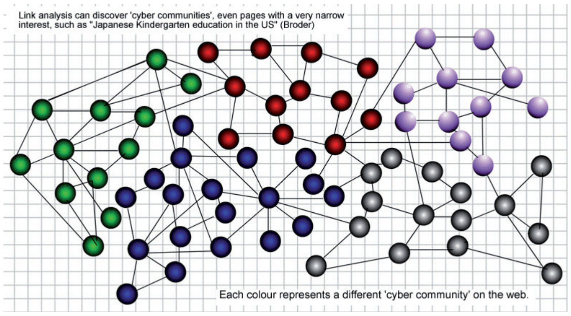
Query independent schemes –a scheme which assigns a score to a page independent of a given query (see PageRank and HITS coming later).

Query dependent schemes – a scheme which assigns a score to a page in the context of the given query (see vector space model).

He then expanded on this: "A query dependant factor is based on the query which has just been made to the search engine. And a query independent factor is a kind of notion of the goodness of a page [calculated pre-query]. He then added how very complicated this type of algorithm is and how they are "changing and tuning it on a daily basis." The use of the word 'goodness' by both Andrei and Brian Pinkerton in my interviews is an important one. Here, they are both using the word in a different context to, perhaps, the way you or I would use it. If we take the word literally, then we could simply be talking about how nice the page was graphically, or how strong the content is in relation to the query. However, Andrei and Brian use it in the mathematical context with 'goodness' being the quality of the algorithm to decide both.

'Cyberspace' (as in the web) already has its communities and neighbourhoods. OK – less real in the sense of where you live and who you hang out with. But there is a sociology to the web. Music lovers from different cultures and different backgrounds (and time zones) don't live in the same Geographical neighbourhood – but when they are linked to each other on the web; they do. Just in the same way as art lovers and people from every walk of life who post their information to the web and also form these communities or 'link neighbourhoods' in 'cyberspace'.

If you read my interview with Andrei, you'll se that, when we are talking about the connectivity server and I mention link popularity, he replies: "It's about link popularity - but much more than that". He quotes how he can find pages of a 'very narrow interest' and map them: "I could find a small community interested in, say, Japanese Kindergarten education in the US, by dissecting the linkage information I can find even these types of tiny communities". It's about as an obscure example as you could make, but these pages could contain information on a variety of other subjects also, including diet, health and social issues for children – but the linkage determines a certain theme or basic connection for that subgroup on the web.


Link analysis can discover 'cyber communities', even pages with a very narrow interest, such as "Japanese Kindergarten education in the US" (Broder)
Each colour represents a different 'cyber community' on the web.

By identifying that type of community, it helps not only in the sociological evolution of the web, but also by providing information on people (in detail) with combined focused interests. This is the 'signature' of a community on the web. Web communities at their core contain a dense pattern of linkage. Here we have thematically cohesive web communities: but not essentially thematically cohesive constrained web sites as in 'themed'.

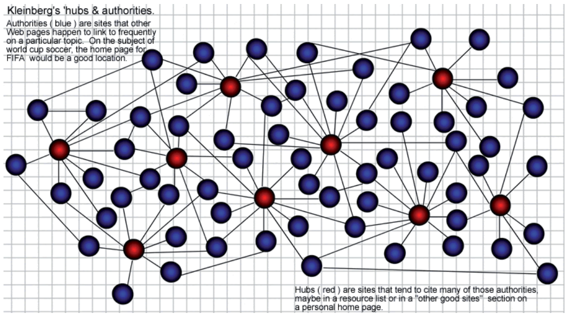
In the main, there are two algorithms developed to 'data-mine' and analyse link structures on the web: HITS [Kleinberg 1998] and PageRank [Brin, Page 1998]. Because these algorithms are so influential I will refer mainly to them.

Few people are cited as often in reference to web link analysis than Jon Kleinberg, Associate Professor of Computer Science at Cornell University, Ithaca, NY (note: the same university as Gerard Salton). His work in the field of information retrieval on the web by attempting to analyse its 'topology' has formed the basis of many other adaptations of his linkage algorithm (including that of Google – his algorithm is cited in the research paper presented by Larry Page and Sergey Brin as well as the foundation work for Teoma amongst many others).

Kleinberg's 'Hyperlinked Induced Topic Search' (HITS) computes what he calls 'hubs' and 'authorities'.

Beginning with a search topic, specified by one or more query terms, the HITS algorithm applies two main steps: a sampling component which constructs a focused collection of several thousands of web pages which are likely to be rich in relevant 'authorities' and a weight-propagation component which determines numerical estimates of 'hub' and 'authority' weights by an iterative procedure. The pages with the highest weights are returned as 'hubs' and 'authorities' for the search topic.

Again, let me try to simplify this: 'Authorities' are web pages with good content on a specific topic. And hubs are directory like pages with many hyperlinks to those pages on the topic. So, a page that points to many others should be a good hub, and a page that many pages point to, should be a good authority.


Kleinberg's 'hubs & authorities.
Authorities (blue) are sites that other Web pages happen to link to frequently on a particular topic. On the subject of world cup soccer, the home page for FIFA would be a good location.
Hubs (red) are sites that tend to cite many of those authorities, maybe in a resource list or in a "other good sites" section on a personal home page.

In its basic principle, this innovation (or expansion on citation and link analysis) is an ideal solution to help ease the problems search engines suffer with mainly text based retrieval, as it works purely on linkage. But applying it to 'Cyberspace' and real world web search has detected its flaws. A lot of further research to 'improve' or 'enhance' the algorithm has been carried out.

Monica Henzinger also gets mentioned a number of times in this guide with reference to her work in the field of web search. At 35 (at the time of writing), she is Director of Research at Google and presides over a group of 10 computer scientists in her research team. A German born PhD she works on improving Google's search functionality and moving Google into new areas such as mobile phone and voice-activated searching. In fact, Google has been approached by the German car manufacturer BMW who want to put a voice-activated search into their 7 series cars – presumably drivers will be expected to stop the car in order to do this and not crash on the highway using a mobile phone whilst viewing a small monitor to check their stocks and shares!

Formerly with Digital Equipment Corporation (DEC) Systems Research Centre, she has conducted much research with other computer scientists (including Andrei Broder, also formerly with DEC Systems Research Centre) into the web's connectivity. She worked with Andrei on (among others) Alta Vista's Connectivity Server project [Bharat, Broder, Henzinger et al - 1999]. This particular project to provide fast access to linkage information on the web, could really be viewed, more or less, as a feat of engineering rather than a feat of science. However, the Connectivity Server provided an ideal software programme to enable 'visualisation' of the web and quickly calculate the 'hub' and 'authority' link based ranking as described by Kleinberg.

Again, the algorithm works like this: given an initial set of results from a query, the algorithm extracts a sub graph from the web containing the result set and its neighbouring documents (those which are linked to and from, creating a graph 'neighbourhood'). This is then used as a basis for an iterative computation to estimate the value of each document as a source of relevant links and a source of useful content. The theory being, following a typical user query, as described, this connectivity analysis should be able to find quality documents related to the query topic. The Connectivity Server provides an excellent visualisation tool for presenting a graphical illustration of the web's 'topology'.

In a further experiment into 'topic distillation' [categorisation and then classification] with Krishna Bharat [Bharat, Henzinger – 2000] they discovered three problems with connectivity analysis as suggested by Kleinberg with this 'links only' approach. The first they describe as: Mutually Reinforcing Relationships Between Hosts. Further described as "where certain arrangements of documents 'conspire' to dominate the computation" (I think we could simply refer to this as 'link Spamming' – 'hub' and 'authority' look-alikes). The second problem they refer to as: Automatically generated Links. This is further described as "where no human opinion is expressed by the link" (think web authoring tools, database conversion tools, or a hypernews system which turns news articles into web pages and then automatically inserts links to the site). The third problem is referred to as: Non Relevant Nodes. Further described as "documents in the neighbourhood graph which are not relevant to the query topic (here they give an example of a query for 'jaguar and car' where the algorithm drifts more towards the general topic of car and returns pages from different car manufacturers as top 'authorities' and lists of car manufacturers as the best 'hubs').

The third problem mentioned, of non relevant nodes is the most common problem when using 'link only' analysis. Which is why it is necessary to also use content analysis in an attempt to keep the computation 'on topic'. By experimenting with 10 different algorithms Bharat and Henzinger were able to achieve considerable improvement in precision. It was also noted that users are perhaps looking more for good 'sites' on a specific topic rather than just a good page.

Connectivity based ranking schemes do help in serving that purpose as many external hyperlinks point back to the root document of a site (home page). Even if it has little content itself, it can usually be the best starting point for exploration (but once again, do not confuse this with 'themed' web sites).

The CLEVER (Clientside Eigenvector Enhanced Retrieval) project, developed at IBM's Almaden Research Centre in San Jose, (of which Jon Kleinberg was a team member as a visiting scientist) uses a version of HITS. Remember that the HITS concept relies on the assumption that if site A is pointed to by many other sites, then they infer authority to A.

However, the definition of 'hubs' and 'authorities' as stated is not very helpful in determining who they are, but you can use an intuitive alternate definition: Good hubs point to many good authorities, and good authorities are pointed to by good hubs.

This "frustratingly circular definition" as it has been referred to as, was solved in the CLEVER project, which used spectral filtering techniques to find the best hubs and top authorities on any given topic. The improved algorithm doesn't merely count links to make its distinctions, it also considers clues within the pages, such as whether the query term is located within or near the link, to ultimately re-rank the original list of sites and present a more accurate measure of relevancy. Users in an IBM-sponsored study found CLEVER's results as good or better than Yahoo!'s 81 percent of the time.

Bharat and Henzinger's work on an improved algorithm has been patented by Alta Vista. The CLEVER search engine is patented by IBM. As such, a licence would need to be granted to any major search service wishing to use the CLEVER technology. In February 2001, Monica Henzinger went on record to say that: "To the best of my knowledge, the HITS algorithm is not currently used by any commercial search product." Even at the time of writing this text [May 2002] I'm not aware of any commercial search service being 'Powered by IBM' or 'Powered by CLEVER'. [Here, it will be interesting to note the patent application by Teoma and how they go about it]

In my interview with Craig Silverstein (Director of Technology – Google) when we discussed the 'meteoric' rise to the top of the search engine charts, he says: "when I joined I knew that Google had a better search technology than other search engines out there at that time". I mentioned at the very beginning of this section that even though all search engines appear to be similar and do the same thing; they are in fact entirely different in the way they do it. This is where Google really stands out from the crowd. To remedy the problem of artificially boosting your connectivity rank by simply getting as many links as possible from anywhere, Sergey Brin and Larry Page (co-founders of Google) created the PageRank algorithm.

Page and Brin describe PageRank this way:

"The reason that PageRank is interesting is that there are many cases where simple citation counting does not correspond to our commonsense notion of importance. For example, if a web page has a link off the Yahoo! home page, it may be just a link but it is a very important one. This page should be ranked higher than many pages with more links but from obscure places. PageRank is an attempt to see how good an approximation to 'importance' can be obtained just from the link structure."

They go on to say that: "PageRank can be thought of as a model of user behaviour. We assume that there is a 'random' surfer who is given a web page at random and keeps clicking on links, never hitting 'back' but eventually gets bored and starts on another 'random' page. The probability that a random surfer visits a page is its PageRank. And the damping factor is the probability at each page the 'random surfer' will get bored and request another random page.

Back to Monica Henzinger here with her official Google hat on to describe the difference between HITS and PageRank:

"The PageRank algorithm differs from HITS in that it computes the rank of a page by weighting each hyperlink to the page proportionally to the quality of the page containing the hyperlink. To determine the quality of a referring page they use its PageRank recursively, with an arbitrary initial setting of the PageRank values. The formula shows that the PageRank of page 'a' – depends on the PageRank of page 'b' pointing to page a [co-citation]. Since the PageRank definition introduces one such linear equation per page, a huge set of linear equations need to be solved in order to compute PageRank for all pages. [Hyperlink Analysis for the Web – Henzinger – IEEE Internet Computing Jan/Feb 2001]

In 1999, Apostolos Gerasoulis, Professor of Computer Science at Rutgers University, New Jersey, became intrigued by CLEVER, Google and the work of the web archaeology team at Compaq's research centre. Whilst working on a research project exploring how to sift mountains of data with supercomputers, for the Defence Advanced Research Projects Agency [DARPA], he sensed a tie-in to search engines.

With his own research team at Rutgers, he developed a prototype search engine called DiscoWeb, a play on the word 'discover' (because it DISCOvers WEB communities – nothing to do with any Saturday Night Fever connotations!). By using link analysis as described in much detail so far, DiscoWeb 'pulls together' highly interconnected web sites that typically share a single topic or focus and automatically builds web directories. Gerasoulis is also the founder of Teoma Technologies, the new kid on the block in the search engine world (at the time of writing). The connection to the work carried out on the CLEVER project is extremely evident even in the name of this new search engine: as Teoma is a Gaelic word for EXPERT.

Teoma uses compact mathematical modelling of the web's structure and its ordering and ranking is based on multi-parametric analysis to achieve its high degree of relevance and quality. In September 2001 Teoma Technologies was acquired by Ask Jeeves. Teoma technology will replace Direct Hit which up until recently powered results at Jeeves and go head-to-head with Google for the title of the web's most popular search engine.

In 2001 another approach to 'fine tuning' Kleinberg's HITS was presented: SALSA (Stochastic Approach to Link Structure Analysis) [Lempel, Moran – 2001] At the time of writing SALSA had progressed from being part 'anecdotal' part 'research', to another ongoing research project with IBM.

For the purpose of being thorough I shall also make reference to 'Hilltop' which is another variation algorithm developed by Krishna Bharat, an expert in the field and a member of the research team at Google:

Just as this edition of the guide was being 'put to bed' as it were, another contender in new generation search engines announced that it had been bought by major player Looksmart. There's a kind of familiar storyline that goes with this. Yeogirl Yun graduated from Stanford with an MS in computer science in 1995. In 1998 he founded My Simon the search and compare price portal which he steered to a $700 million acquisition by CNET. He founded Korea-Wisenut in 1999 and then Wisenut in 2000. The acquisition of Wisenut by Looksmart in a $9.25 million stock deal provides Looksmart visitors with both directory and context sensitive results. Wisenut's patent applied for technology works on an expert pages link and link anchor text analysis detail in a similar way to that of Teoma.

Before I move on to another and most important aspect of link analysis, let's take stock of where we are at this point and look at the initial benefits that these types of connectivity algorithms provide to search engines.

First: There are examples to be found as covered in the co-citation method illustrated earlier. Given the web graph used by a search engine it can be analysed to take a co-citation example in the way that Andrei Broder explains: "Let's take a page about London. And here is a good list about Hotels in London. So what we get here is the notion of endorsement which is captured by the link. The other notion is of sites being related, I mean, typically, the fact that, immediately after a link, for, well I'm not too familiar with London Hotels, let's say Grosvenor Palace follows a link to, say, The Four Seasons Hotel, this indicates that those two things tend to be related. Because, on a page, close to each other this is what is called, technically, co-citation, in other words - someone who cited those two places, that is then an indication that these two things are related, they probably follow the same kind of theme." (here, again, theme is used in the sense of classification). "So, what we have here is that, a frequent co-citation suggests relationship when the co-citations are close together on the page. So, pages with many co-citations which occur close to each other on the page containing the co-citation tend to be related."

Again, in the Spam battle, it is also then possible to follow the 'path' of a URL and look at its similarity. Mirrored hosts are inevitable on the web and have already been covered in this text. The path of an URL following the host name comes after the third slash. For example:

http://www.searchengine-report.co.uk is the host and

http://www.searchengine-report.co.uk/how_search_engines_work.html

is the path. Two hosts, let's call them 'me.co.uk' and 'myimage.co.uk, are mirrors, this occurs only, and only if, for every document in each host with the same path which is a similar document they have the same path and vice versa.

> The structure of the web and its 'connectivity' provides many clues to search engines for what could be deemed as more important pages.

Categorisation and classification has been touched on a number of times during this section of the guide (and will be even further), for this reason hyperlink analysis is used by all search engines to compute statistics about groups of web pages. They can gauge their average length and the percentage of terms which are in a different language etc. etc. And they can also determine the number of pages (categorically) in a certain domain i.e. .com, .co.uk etc. (I mentioned the 1999 experiment earlier in this section which discovered that 47% of all web pages at that time belonged to the .com domain).

In terms of geographical scope, whether a web page is only for people in a given region or is of national or international importance can also be discovered.

If a TV listings page is interesting only to the region it covers and income tax is important to, say, UK expatriates worldwide, then this community can be 'flagged up' by its linkage and topic. There are many things to be learned by 'connectivity analysis' of the web.

In the conclusion to this section I'll be touching on 'learning machines' (vector support machines) and artificial intelligence (AI) which is where the field of web search and retrieval inevitably has to go next. First, I want to look at the importance of hypertext writing and then tackle the final part of this section of the anatomy of a search engine with the query interface.

All of the major search engines place a great deal of importance on link anchor text for their retrieval algorithms. Latent semantic content (as it's known) provides the real key to what other pages say about your own.

As you are now aware, the structure of the web and its 'connectivity' provides many clues to search engines for what could be deemed as more important pages (both for crawling purposes and ranking). However, there is more to the structure than just the hyperlink connectivity patterns: Anchor text.

Connectivity based algorithms can give a basic insight to perhaps the inherent value of a document via its connections: But does that really mean 'quality'? What defines the quality of a web page? Well this is a matter of human judgement: Can machines gauge it? There is little empirical evaluation of link popularity algorithms to prove that this type of citation – co-citation analysis correlates to human judgement of quality. [Amento, Terveen, Hill - 2001]

However, given that it's perhaps easier for a machine to get an idea of what a page is about by what other pages say about it than what it says about itself, then the closest approximation of topic is likely to come from the anchor text in the pages which point to it. Not only does anchor text provide clues to what a page is about, it also helps to return to the search engine index details of pages which cannot normally be indexed such as images, programmes and databases (see section on problem pages). In fact, link anchor text can provide details about pages which have not yet even been crawled.

From time-to-time, Google will return a list, or even a single link to a document which has not yet been crawled but with notification that the document only appears because the keywords appear in other documents with links which point to it.

All of the major search engines place a great deal of importance on link anchor text for their retrieval algorithms. Latent semantic content (as it's known) provides the real key to what other pages say about your own.

The use of link anchor text by a search engine was incorporated into the results of WebWorm as far back as 1994. [McBryan 1994] And further back in 1992 Frei and Steiger described a way for using semantic content of hypertext links for retrieval. Latent semantics can provide many clues for machines as to what a page is about. However, even this information, at times, can confuse a machine due to the way we write for the web.

In her dissertation "The Importance Of Being Different" [Emitay 1997], Einat Emitay, whilst at the Centre For Cognitive Science at the university of Edinburgh, cited the work of Dillon et al [1993] on the subject of 'schemata' or 'genre conception' when writing for the web.

Dillon had discovered that a problem existed in hypertext because of the flexible nature of language and the varied layout used in its creation. After analysis of the way hypertext documents are written Amitay's dissertation describes the linguistic conventions with which hypertext documents are written. Her discoveries have added a lot to the importance of link anchor text in the field of information retrieval on the web.

As a PhD student at the Division of Information and Communication Sciences (ICS), Macquarie University, Sydney, Australia on scholarships from the Microsoft Research Institute and CSIRO Mathematical and Information Sciences, her main research interests have been carried out in the field of statistical NLP (Natural Language Processing), information retrieval and extraction from hypertext, and defining language use conventions in hypertext documents. Her work attempts to combine IR & IE techniques with language patterns found in web documents. She has developed a tool for automatically collecting and filtering descriptions of web pages which is called InCommonSense. The principle behind the use of InCommonSense (which was presented in the paper: Automatically summarising web sites – is there a way around it? – 2000) is the use of link anchor text to create the descriptive summary which search engines provide in their lists following a keyword search. Basically, Emitay says that it is more effective to use a synopsis of what other people say about your site in their link anchor text than it is to use a webmasters own description taken from meta tags or just a snippet of text surrounding the terms in the query in the way that Google does.

Winner of the somewhat 'conspicuous' title 'Sexiest Geek Alive' 2001 award, Ellen Spertus has also conducted an enormous amount of research into the usefulness of link anchor text in information retrieval. Ellen is Assistant Professor in the Department of Mathematics and Computer Science at Mills College, Oakland, California.

She has also worked at Microsoft. In 1999 she gained her PhD at Massachusetts Institute of Technology (MIT) with her thesis: ParaSite: Mining the structural information on the world wide web and introduced SQUEAL, a structured query language (SQL) for the web based on searching semi structured information including hyperlinks, structure within hypertext pages and structure within URL's.

I've mentioned and referred to many of the most influential figures involved in the research and further development of search on the web. One person in particular has been cited many times because of his work, yet I have not given a proper endorsement and credit to the massive amount of work he has put into the field. Not only is he responsible for some of the most credible work, he is also teacher, colleague, mentor and friend to many of the other people cited in this text. He is assistant professor to the Computer Science and Engineering Dept, Indian Institute of Technology Bombay.

Before that he was at the IBM Almaden research centre where he worked on hypertext databases and data mining.

Soumen Chakrabarti has made a number of comments which have done what a good comment should do – make me think – or make me laugh. I laughed when I saw that he had spent enough time to discover that his name is also an anagram of 'anarchism outbreak'. I also laughed when he described 'link spamming' for what it is: 'A nepotistic clique attack'. A wonderful description of nothing more than 'fake links'. But then with his academic and professional 'hat on' he describes this 'Spamming' effect as a collection of sites linking to each other without semantic reason. In another section of this guide I'll be covering link Spamming, but more to the point, explaining the poor consequences of FFA (free for all) sites and you being a part of them.

Since 1997 Soumen Chakrabarti has been working on machine learning for the purpose of effective information retrieval from large hypertext databases. In a very interesting analogy he describes the phenomenon of IR on the web as thus: "The web grew exponentially from almost zero to 800 million pages between 1991 and 1999. In comparison, it took 3.5 million years for the human brain to grow linearly from 400 to 1400 cubic centimetres. How do we work with the web without getting overwhelmed? We look for relevance and quality."

He was leading member of the CLEVER project team and has developed three systems to further advance the relevancy factor in web search. They are: A focused crawler which can build a topic-specific library by crawling a negligible fraction of the Web. A hypertext classifier that analyses the text in, and links around, a given web page and automatically assigns it to suitable directories in a web catalogue such as Yahoo!. A popularity rating agent that analyses the link around a web page and the text in pages that cite the given page to assign a measure of popularity to the given page.

His most recent work (at the time of writing), published in 2001: "Integrating the document object model with hyperlinks for enhanced topic distillation and information extraction" is centred around better topic distillation, web search using devices with small or no screen, focused crawling, annotation extraction and data preparation for linguistic analysis. His book on mining data from the web is due to be published mid 2002.

This roll call, as it were, in the field of search technology and information retrieval, would not be complete without a quick profile of Hector Garcia-Molina whose work is also cited a number of times in this text.

Hector Garcia-Molina is the Leonard Bosack and Sandra Lerner Professor in the departments of computer science and electrical engineering at Stanford. In January 2001 he became chairman of the computer science department. From 1997 to 2001 he was also a member of the president's information technology advisory committee.

His research interests include distributed computing systems and database systems. He is a Fellow of the ACM, received the 1999 ACM SIGMOD Innovations Award, is on the Technical Advisory Board of eGuanxi, Enosys Markets, Maaya, Metreo Markets, Morhsoft, Radik, Times Ten, Verity; and is a member of the Oracle Board of Directors. You need only look at his credentials to get an indication of both his status and the influence of his work in the field.

I mentioned Cyber Communities earlier in this section and quoted from my interview with Andrei Broder ("Japanese kindergarten education in the USA").

Taking into account all of the previous work in the field of link analysis on the web, another group of researchers published a study in March 2002 called Self-Organisation and Identification of Web Communities [Flake, Lawrence, Lee Giles, Coetze] which highlights the problems with both HITS and PageRank in the sense of being able to determine a deeper analysis of the organisation and sectors of society on the web. By using what they term as an 'approximate community algorithm' they have been able to find and rank online communities to prove that, even with billions of pages, we can, in fact, organise ourselves online.

There is one other area of note in search engine technology which does not concern itself with indexing text or analysing links, but it does merit a mention here whilst covering algorithms, heuristics and innovators.

As a student at MIT, Gary Cullis entered the $50k business plan competition (affectionately known as the "I Wanna Be a Gazillionaire Geek").

His idea was spotted on the MIT website by Mike Cassidy, an entrepreneur looking for a new start-up to get involved in. The business plan won joint first place in the competition and Direct Hit was born. Direct Hit technology is based on human behaviour and feedback i.e. it tracks user patterns following a search query and monitors which sites are clicked on most following queries and how long people stay on those sites. Direct Hit then uses this information to return the most popular sites as (seemingly) voted for by surfers. The first 'partner' search engine to adopt the technology was Hot Bot. Ask Jeeves acquired Direct Hit and used the technology on its own sites, however, in a statement in 2002, Jeeves said it was dropping Direct Hit for Teoma (which it now owns).

If you read my interview with Andrei Broder, you will note that he says: "We do use this type of tracking data at Alta Vista, as all search engines do, but at Alta Vista we don't use it for ranking purposes."

In this section covering retrieval and ranking algorithms and heuristics it's plain to see that the link structure of the web (insofar as the partial content of the web which each search engine has indexed) and the work of Jon Kleinberg with HITS, provides the most important forward motion in this field of research. Yet, because of the exponential growth of the web, even hyperlinked topic distillation of the web fails for a number of reasons.

It's a step forward, for sure, but even with the massive amounts of research it's more-or-less a 'baby step' when compared to the challenge of scaling up to not just the millions and millions of existing and new pages on the web, but the billions and billions of links which have to be analysed in order to make sure that the minute number of 20 pages (the top ranked pages following a query) are the most relevant to return.

As to how accurate PageRank and HITS type algorithms can be, remains to be seen. Google has already suffered a number of slightly embarrassing situations in the course of 'tweaking' their algorithm. In November 1999 Danny Sullivan reported in his Search Engine Report newsletter that, it had been discovered by a poster to an online forum, a search at Google for 'more evil than Satan himself' actually returned the Microsoft web site at the top of the results. What could cause this? Well, most likely that the word evil appears in the link anchor text, or the text surrounding links on many pages pointing to the Microsoft web site.

Once again, in February 2001, Danny Sullivan reported that a search on Google for the 'particularly insulting phrase' 'dumb Oedipus' the top result was the official George W. Bush campaign store. The reason for this was a link from a men's satirical web site which had a link to the Bush campaign site and the insulting words close to the link on the page. This in itself was enough to push the Bush page to the top for this most obscure search.

There are other occurrences of this type of embarrassing scenario with other search services, but I'll leave it with the Google examples for now.

Are link based ranking methods better than content based? It interesting to note that, TREC (text retrieval conference) which has mainly concerned itself with subject searches has recently expanded to new search tasks such as question answering and new document sets including several web collections. New experiments by TREC suggest that link based ranking methods are actually no better than traditional content based methods [Crasswell, Hawking et el – 2001]

THE QUERY INTERFACE

I want to move on now to the final component in my attempt to break down the anatomy of a search engine. Given that you've been able to understand a lot of what has been written in this section so far, then you'll be aware that it's here at the query interface where the whole thing has to come together.

When the French author Victor Hugo, had Les Miserables published, he was not living in Paris at that time. He was waiting to hear news from his publisher about the kind of reception his new book was having.

When he could wait for news no longer, he sent a letter to his publisher which contained only the character:? On receiving this, his publisher knew exactly what it meant and he returned a note to him containing only the character: ! This let Victor Hugo know that his book was a huge success. It is said that this is the shortest correspondence in history.

Because the two men knew each other so well, and understood the context, they were able to have a meaningful understanding of what the symbols had encapsulated in this briefest of information exchanges.(purely for information - Les Miserables also contains one of the longest sentences in French at 823 words before a full-stop/period)

Why does this minor episode in history merit a mention here? Well, the average query at a search engine interface usually consists of no more than two to three words. And from this tiny fragment of information, a search engine is required to return relevant documents from the millions and millions it has indexed.

In the paper 'Analysis of a very large Alta Vista query log' [Silverstein, Henzinger et al – 1998] they presented an analysis of six weeks worth of Alta Vista user queries. By doing this they were able to create a model of user behaviour by monitoring the 285 million queries. Not surprisingly, the facts suggested that web users differ significantly from those involved in the field of information retrieval. Users mostly only look at the first ten results and rarely modify their query. In general: users are lazy. Another phenomenon was the number of users who type the URL of the site they want to visit directly into the search box instead of the address bar at the top of their browser.

> The average query at a search engine interface usually consists of no more than two to three words.

However, the most important element I noticed in the paper was the table covering the 25 most searched for terms during that period. I'm not at liberty to publish the list here, but I can tell you that, the most frequently searched for term was 'sex' at almost 16 million and that 15 out of the top 25 were sexually oriented search terms. I looked at a more recent report [2002] and discovered that since this analysis in 1998, only Spice Girls needs to be replaced by Britney, and Titanic by Lord of the Rings (at the time of writing): and the list remains virtually the same.

User behaviour around the search engine interface can give a lot of clues to search engines on the most popular queries, so search engines should really be able to cache the results of all of the most popular queries. In this way, a search engine could have popular queries pre-processed i.e. the best sites for the query Titanic, but this then leads to a temporal tracking problem in that, while the movie was popular at the time, it is much less so now (the word temporal is sometimes used to describe click through analysis in the manner used by Direct Hit but this a very loose use of the term).

The problem of short queries is very much non-trivial for search engines, as we already know. A search engine really has no idea of the context or classification of your search.

The way that results are presented at a search engine interface and the reasons for the selection are determined by the many factors I've already covered in this section. Each search engine has its own 'special sauce' made from the various ingredients.

Search engines are striving for better ways to deal with short queries. In the section on link popularity I'll be touching on the connectivity server and Inktomi's web map project.

I mention it here though because of some relevant comments about short queries made by Eric Brewer, founder and Chief Scientist at Inktomi. Inktomi's web map project provided an excellent opportunity for them to 'clean up' their database of 1 billion documents at the time. The web map was able to show duplicates and relationships between documents. From this they were able to pull out the Spam which made up much of the database and get a cleaner and easier to handle 500 million documents. But even bringing the size of the searchable database down this far, Brewer himself says: "Now we've got 500 million documents in the database. But you have a two word query, and we're supposed to give you an answer."

What Eric Brewer believes is the answer for Inktomi are what he calls 'context zones'. He's further quoted as saying that context should be able to change on the fly, based on the query words and gives the example that some words are magical because they imply intent, and intent should change the context.

Think here again about the classification of search as I covered it earlier in this section.

Eric Brewer's example goes like this: a search on flowers might bring up documents on Roses, links to horticultural and other sites. But the query 'buy flowers' signals intent and the most relevant results should lead to links to florists [Sherman – About.com] As you can tell from that, it's about an effort to identify the 'classification' or 'nature' of the search in order to provide the most relevant results. But again, please don't confuse this with what some search engine optimisers refer to as 'themed sites' – as you know, this notion is dispelled both here and in other sections of the guide.

[NB – At the time of publishing this second edition of the guide, Inktomi had just been granted a new patent: Method and apparatus for retrieving documents based on information other than document content.]

Eric Brewer may call them 'context zones' but the search engine Northern Light has tried something along these lines with its special folders approach (another patented technology for 'distillation and document clustering'). The intention, of course, is to provide a way of automatically presenting results in a categorised, summarised and organised fashion at the query interface.

So how does a document clustering method perform against a single results list set? Let me give another quick example here of just how wide from the mark a search engine can be. The various use of stop lists and word stemming can affect the results to certain queries (depending on the search engine).

Even super computer systems like that of Google are 'blind' in the sense that they can't read web pages. They can take all of the similarities of text as a digital computation: but they can't really understand context and semantics (cast your mind back to when I covered the classification of search in this section of the guide).

If I use the search phrase 'pictures of the Madonna', Google is unable to determine if I'm looking for pictures of a pop star or pictures of a work of art. As both 'of' and 'the' are stop words at Google, using the above phrase brings back the pop star 'Madonna Photo Album' page at number one and 'Sexy, Sexy Pictures' of Madonna at number 2. [Google – 07/02/2002] To get pictures of 'The Madonna' a user would need to understand Boolean logic, or at least include, perhaps, Michelangelo somewhere in the text string in order to create the differentiation. If Google was able to determine the semantics and had a form of automatic classification, then the user would have a display returned which gave some alternatives: all pop music fans this way: all art lovers this way. Unfortunately, at this time, this mindless machine is unable to determine the importance of a pop star over that of the image of the mother of Christ.

Try the same search over at Northern Light with its document clustering algorithm and what do you get? Well, there is a mixture in the main set of results of both 'The Madonna' and 'Madonna' the pop star. However, at the top of the custom folders section, you do at least get 'Museums & Galleries'.

Now, Teoma, with its research and background in link analysis, provides authoritative web pages, finds links to experts (or should that be teomas!) and then groups results by topics. Does it work? Well, Teoma was Beta testing at the time of writing, but using the same query it was able to return a Madonna the pop star at

1 and Michelangelo pictures at 2. Not a bad shot. And in the grouped by topic section there was a folder for religious posters as well as a folder for pictures of Madonna the pop star. This is most definitely the way forward for presentation of search results closer to being in context.

[NB – Northern Light has been acquired by Divine Inc. a provider of content management and delivery solutions for enterprise customers. At the same time Yahoo! announced a partnership which allows them to provide access to Northern Light's Special Collection of over 70 million pages under the banner of 'Yahoo! Premium Pages'. Surfers can search the database, view a summary and decide whether they wish to pay the fee to download the document – same now applies at the Northern Light interface.]

Search engines are racing to find ways to provide a better user experience by using the many techniques described in this text, to provide, fresher, more relevant documents following a query. The better then the experience should be for the end user.

As you are aware by now, a 'text based only' or 'content only' retrieval system falls down in many ways, but mainly because of ambiguity and naivety. The clues to the most relevant documents following a query may well be 'off the page' i.e. in connectivity patterns and link anchor text. However, providing results which are deemed to be the most relevant, even if the query string does not appear in the summary of the results, can also cause confusion and frustration with the casual surfer. And for the more 'sophisticated' searcher who really wants to have control of the whole thing – widening – narrowing – it's actually a bit of an insult, because they know exactly what they want and how to get it (if it exists at all in the database).

Search engines know how difficult it is to elicit a better query/question from the surfer – if the surfer knew how to do it then they would. Trying to strike a happy balance between helping casual surfers enjoy a better user experience and let the professional searchers maintain some sort of control over the way that results are presented is not easy. The use of query expansion techniques and re writing queries on the fly in the hope that this will return better results are used by all search engines. But even Monica Henzinger of Google goes on record to say that: "We can't completely rewrite the query into something that we think is more appropriate, because, you know, people like my husband would get crazy. He just wants to find pages that have his words." [salon.com – June 2001]

I've cited the work of, and quoted from the work of, a number of the leading authorities in the field of search technology and information retrieval, and when it comes to the results as seen on a page following a query, there is one person who has specialised in the field of optimising search by showing results in context.

Susan Dumais' interest is in algorithms and interfaces for improved information retrieval, as well as general issues in and human-computer interaction. She joined Microsoft Research in July 1997 and works on a wide variety of information access and management issues, including: text retrieval and categorisation, collaborative filtering, interfaces for combining search and navigation, and user/task modelling.

It's very interesting, in that, before she joined Microsoft (prior to 1997) she had already worked on a statistical method for concept-based retrieval known as Latent Semantic Indexing (already mentioned in this text) with Bellcore, now Telcordia. Her work is now patented by Telcordia. Basically LSI is a means of finding relevant information following a query, even when they do not share the words of the query.

As I conclude this section on the anatomy of a search engine, you will now be aware of the enormous complexities involved in the whole process of what appears to be the simple task of, say, returning relevant results for 'green card' – would that be a query re immigration documentation?Or would that be about packaging material? The fact of the matter is, if you could take all that cluttered data on the web, crawl it extensively and then create a structured taxonomy of pages which exist to provide relevant information in the way that Yahoo! and other directories do using humans, but do it automatically – semantically - categorically – hierarchically -regionally - then a search engine really will have the Golden Goose.

THE FINAL HEURISTIC

I've covered a great deal of what happens with search engines using advanced, automated information retrieval techniques. But there is one final ingredient to add. Sometimes when you do a keyword search at a search engine and analyse the results, there seems to be no reason why certain pages achieve such high ranking. The keyword density for that engine may not be quite right and the linkage may be very poor: In fact some sites may have no linkage at all. I want to point you to my interview with Brian Pinkerton again. Note in the interview how he mentions the use of 'editors' with WebCrawler and Excite. Also note in my interview with Martijn Koster, how he also mentions editors. Brian states: "One of the ways in which search engines achieve… well I know some search engines do this, but I don't know about all of them, but they do actually editorially determine the results for the top, say, thousand queries. Somebody will sit down and say, you know the query… travel… is very popular… and you know, if you run a query like 'travel' on any search engine (even Google) the results are essentially meaningless.

So why not have an editor just whack out 25 good travel sites and stick 'em at the top!"

Do all search engines really do this? I think there is most definitely a little of the 'human intervention' which goes into the algorithm sauce at every search engine. A former engineer at HotBot was happy to say that, yes it does happen. Think about the way Ask Jeeves has its editorial team thinking up answers to all of those queries by using human judgment to decide on the best results for popular queries, and then ask yourself: why would any other search engine not add a little of this for quality control purposes.

~ End ~

THE INTERVIEWS

Overview

I've been very fortunate, in that, I've had access to so much information for this edition of the guide. I've also been very lucky in being able to get to speak directly with so many leading figures from the industry. Some of the following interviews were carried out face-to-face and some by telephone. As a former professional broadcaster, I really enjoy doing interviews. I'd like to personally thank all of the people in the list below for giving me their time and for allowing me to 'pick their brains'. I'd like to say a special thank you to Andrei Broder (Alta Vista). Although I was only scheduled for a 15 minute slot, we actually talked for almost an hour. It was a most illuminating conversation and Andrei is a very funny guy. Another special thank you, goes to Brian Pinkerton (WebCrawler/Excite) for keeping me right on a number of issues and pointing me in the direction of so much useful material. As developer of the web's first full text retrieval search engine, Brian has his own place in Internet history. It was an honour to have the support of someone whose contribution to the science of information retrieval on the web, has been so important. And thank you so much to Craig Silverstein, Director of Technology at Google for his extremely helpful insights into best practice SEO.

The full transcripts of my interviews are included as follows:

Andrei Broder

Andrei is VP Technology and Chief Scientist with Alta Vista. He is largely regarded as one of the world's leading experts in information retrieval and search technology. He is the author of many research papers for which he has won awards. In this interview, Andrei explains the characteristics of search as viewed by a search engine, discusses the Term Vector Database, The Connectivity Server and gives a hint to the new digital camera he'd like me to buy him as a present!

Mike: Hello Andrei

Andrei: Shoot… [Laughs]

Mike: [Laughing] Good I can tell that you're ready for this!

Mike: I must do the background first as usual Andrei. You're formerly Compaq Systems Research Centre, and now Vice president for Research and Chief Scientist with Alta Vista – correct?

Andrei: Correct!

Mike: So – tell me – what does the job of Chief Scientist involve on day to day basis?

Andrei: Well, basically, I have the responsibility to determine the search technology that we adopt and what type of things we do, and where we put our efforts. Coming up with ideas and trying those which are floating around. Some are good – some are less good. And we need to pick and choose which to use.

Mike: The main reason for wishing to speak to you Andrei, is to find out a little bit more about the technology that you're using. In the sense that, there's so much information out there on the web, just how accurate, or more to the point, how relevant are the results that we get when we perform a basic key word search?

Andrei: Right. Let me give you maybe some more background before we proceed. Web search is different from classical information retrieval, at least among people on the scientific side. I think this has become quite clear. And the way that I characterise the type of searches people do is roughly three wide classes. The first class is really informational. It's when you really are looking for a piece of information on the web. So we make a query like say...

'low haemoglobin' for instance. This is a medical condition. You are looking for specific information about this condition. That's very close to the classical information retrieval. The second class, which is very specific to the web is what I call 'navigational'. And navigational is when you really want to reach a particular web site. If you do a query like... say United Airlines for instance. Probably what you really want is to go directly to the web site of United Airlines – like www.ua.com just like if someone types BBC, it's most likely they want the web site of the BBC and not the history of the BBC and broadcasting or something like that. They probably want to just go directly to the web site. Those are what I call navigational queries and I guess we all do a lot of those. What we see from analysing our logs is about 20% of what we believe to be navigational queries. The third class is transactional. Transactional means that ultimately you want to do something. Something on the web, through the web. Shopping is a good example. You really want to buy stuff. Or you want to download a file, or find a service like, say, yellow pages. What you really want to do is get involved in a transaction of information or services. Take a shopping query like... Sony F707 which is a camera from Sony...

Mike: [Laughing] Yeah, of course, I knew that...

Andrei: [Laughing] Well... if you're thinking of a present for me... anyway... those are transactional queries where people want to buy stuff and so on. So, they are wanting a return which needs to satisfy this need. So, I think it's important when you're talking about relevance and precision to distinguish between these three classes. Because, for instance, for the classic transactional query, with me living in California, it's likely to be something different to what you want living in the UK. So what's happening with transactional queries, it's difficult to decide what the best result is, you know the context plays a big role. And in fact, often with this type of transactional query, the traffic from other sources, is often better than the what we collect. It's often more up to date or it's more appropriate because it's a pure shopping query, you know when you go shopping, you better be in a shopping mall – not in a library [Laughs].

Mike: So anyway, Andrei, with millions of documents returned on certain keyword searches, after, well, could it be the first couple of pages or so - the relevancy factor must become fairly vague after that mustn't it?

Andrei: Yes, it is. If it's an informational query we're talking about then, yes, the relevance drops quite rapidly after the first few pages. And the main reason we say that, you know, we found this many pages is not because people can, make use of that directly, but because advanced searchers like to know the size of the results because they make complicated queries and they want to, usually, narrow the query. So if you make some changes, you might be interested to know, did I narrow the query, or didn't I. For instance, you might make a query, like we said before, 'low haemoglobin', so let's stay with that. Let's say you do something like, 'low haemoglobin' and 'anaemia' you will get a different result. You might want to just remove the word low and have just 'haemoglobin' and 'anaemia' to increase the query. You want to figure out whether you are narrowing the field, or are you expanding it. So that's mostly the reason we do it – for advanced searchers.

Mike: So it is very much a case of providing all of the results more for advanced searchers?

Andrei: Yes very much so. Frankly, I believe that your average searcher doesn't do this. You know, if you make a query like for IBM, you're gonna have millions of results. The important thing is, even though IBM is more of a navigational query, you'd better have www.ibm.com as result number 1.

Mike: Is it fair to say Andrei, that if there's a community of search engine optimisers out there, which there certainly is, and they're trying to fly their pages to the top of the results, then some of those pages have a certain unfair advantage. I mean that, a lot of, perhaps, more relevant pages which are not optimised in that way may never actually see the light of day.

Andrei: Well, the risk exists there of… well, you know, the search engine optimisers will tell you that they can do everything. There are a couple of things here. One is that, in any marketing environment there is a certain amount of advantage. I mean if you have enough money to buy a full page advert in The London Times, you're gonna get more exposure than someone who doesn't have the money to do that. It's a fact of life that to some extent, money will buy exposure. So you can spend money directly buying an ad for exposure, or you can spend it indirectly by making a more attractive site, more friendly to search engines than someone else does. There are certain techniques. I mean there are ethical search engine optimisers and there are some which are less ethical. The methods and techniques that the less ethical ones use, for a lot of them, it's funny because this can easily be turned against them. Without realising it, they can create a certain kind of signature, for a site that's quite visible to us. Like creating a lot of fake links for instance. Well these can fairly easily be determined. And from time-to-time when we decide, well this practise is very annoying, it's really interfering with our results then we unleash some process and that's the end of it for those guys. They're dropped completely from the index and it's because of the fact that they're using fake links or whatever that they suffer the penalty factor. Strange thing is, for some of these unethical guys they don't mind this happening [bursts out laughing] they just go back to their clients and say: "You lost your rank I'm afraid" and then charge the clients again! $300 per URL sometimes because they dropped!

Mike: [Laughing] And there are some mugs out there paying it too! But the fact is that, you can spot a pattern.

Andrei: Exactly!

Mike: Let's go back to the original example I gave you Andrei. Let's say, someone like yourself, a scientist, had written an authoritative paper on a particular subject and you don't know anything about optimising for search engines and simply post it on the web, say it's information about a drug, or something. And then somebody performs a keyword search on this subject, they're more likely to get a commercial site than the academic…

Andrei: Yeah, but… the restriction is extremely small… when you're talking about people who do search engine optimising. You know, I can figure out that a junior faculty member is not paying a search engine optimiser so that his paper comes first [Laughs] I don't think this happens very often [Laughs again]. You know, more search engine optimisation happens in the commercial sector, where we see a lot of abusing, a lot of competition, with the gambling sites or real estate for part time Condos in Maui this kind of thing. So typically, if you are doing scientific searches you'll see hardly any interference at all. It's extremely unlikely that a search engine optimiser will try to get… you know, what you're talking about, like a medical paper. Let's say you write a paper about an advanced form of anaemia say, well a company which makes say, herbal products, it's less than likely that they'll really go and work on these types of scientific terms, it's the commercial terms that they'll be working on. Because, obviously, someone who is looking for articles of this type will not be looking to the market for herbal remedies, or Viagra or something [Laughs] this kind of stuff, so it's a mutual interest really. It's exactly the same reason that people don't put ad's for dishwashers in Extreme Ski magazine!

Mike: [Laughing] Yes – interesting marketing concept that one Andrei! OK Andrei, thanks for talking me through that. But the main reason I wanted to speak to you is about a couple of high level projects you've been involved in over time, which I've looked at. Not being a scientist like yourself, I have to say, I found this to be the most difficult part of my research. For instance there is the Connectivity Server and also the Term Vector Database. I know this is asking a lot – but is it possible for you to give me a very general explanation of both and how they've had impact, if any, on the future of information retrieval on the web.

Andrei: Sure, sure. Well, first, the Connectivity Server is a project where we tried to put all the linkage information data on the web, which is to say which page points to which page, in a format so that we can very easily analyse it.

Mike: So, is this basically about what's commonly referred to as the 'link popularity' factor?

Andrei: Well, it's link popularity, but it's also a lot more Mike. There's been a tremendous amount of research here, as well as elsewhere, just about every commercial search engine, but also in academia, in IBM research and so on about the use of linkage information. You know, what's very interesting about the web is the hyperlink environment which carries a lot of information. It tells you: "I think this page is good" – because most people usually list good resources. Very few people would say: "Those are the worst pages I've ever seen" and put links to them on their pages [both Mike and Andrei have a good laugh at this idea]. OK – look at it this way, let's take a page about London. And here is a good list about Hotels in London. So what we get here is the notion of endorsement which is captured by the link. The other notion is of sites being related, I mean, typically, the fact that, immediately after a link, for, well I'm not too familiar with London Hotels, let's say Grosvenor Palace follows a link to, say, The Four Seasons Hotel, I don't know if you have a Four Seasons Hotel in London [Mike has a little chuckle here at the notional thought of a royal palace pointing to an hotel] this indicates that those two things tend to be related. Because, on a page, close to each other this is what is what is called, technically, co-citation, in other words. Someone who cited those two places, that is then an indication that these two things are related, they probably follow the same kind of theme.

Mike: But the theme is not the same as used by search engine optimisers is it? I mean, it's about linkage data and co-citation?

Andrei: Yes, absolutely. Say for instance, I will see on many, many sites, I will see a particular hotel in London which has a link to another particular hotel in London, and then I can say that those hotels are related, that somehow they form the same notion. So, these are the two main meanings attached to links. If you have access, fast access and efficient, it tools the entire graph with this link, you know, to the billions and billions of links there are. And roughly, there are about ten links per page, so we're talking about a web of about a billion pages, so we're talking about ten billion links. It's very non trivial how to represent and how to deal with such a large amount of information. But assume we can do that, then we can a do a lot of interesting analysis, in particular, as we were talking about Spamming, we have very sophisticated methods to determine link Spamming so that I cannot be fooled. We don't always want to apply this methods, for various reasons, but we do have the technology to determine link Spamming very well. Let me give you an example of the kind of analysis which can be done. People do a lot of stuff with this linkage information. It's possible to find 'communities', very small communities of people linking to each other with pages of a very narrow interest, like I could find a small community interested in, say, Japanese Kindergarten Education in the US, by dissecting the linkage information I can find even these types of tiny communities. There are lots of interesting things which can be done which is why we wanted to have this connectivity database of linkage information and a very convenient way of attacking it. And we did some very interesting studies about the actual shape of the web, you know, how it looks as a graph. We also did a lot of internal studies, which we cannot publish for obvious reasons.

Mike: So it was, basically, creating a kind of roadmap of the web.

Andrei: Yes, it was getting an indication of how the web looks as a graph. If you ignore the content and think of each page as a dot, and each page has arrows pointing from it which represent the links. So if you have a page about newspapers in London, then you probably have arrows which point towards The Times and The Sun and other English newspapers. So looking at that, and forgetting about content, you can do an enormous amount of analysis. And all search engines do some kind of linkage analysis in some form or another. This is absolutely necessary in this hyperlink environment. Let's take the navigational kind of query that we mentioned earlier. Let's say you want to search for United Airlines and you really want the .www.ua.com site, the only way really to make sure of that is to take the pages which match that query for the words United Airlines, which could be thousands or even millions, and lots of people wrote, in the anchor text of the link: united airlines. So this tells you that they are pointing to the site of United Airlines.

Mike: So that's about the relevancy of the link. I think the search engine optimisation companies realise now that the 'link popularity' factor and the link anchor text is important...

Andrei: Oh absolutely, absolutely. But the point is that it's very easy to say, well I don't want to go into technical detail, because I can't reveal too much about exactly what we're doing. But just to give you a 'hint' of it. It's very easy to create a thousand fake domains and then they all point out to my favourite page: OK. But this is level one. The problem is that, your thousand fake domains don't have any links to them! So it's easy to say: "hey this doesn't look right".

There's a thousand pages in a thousand domains each one pointing somewhere, but nothing pointing back. So, again: OK. But now, if you're thinking smart, you realise this yourself, so you start making links back to them – only now, you have to understand and know an awful lot about the statistics of the links to create a fake that's believable. It's just like faking a master picture, you know, you have to be a very, very good painter to be able to create a fake that would fool an expert.[Laughs] And for the web, where you have so much data, you don't have to fake a picture – you have to fake an entire museum! [Laughs]. You see I don't concentrate on looking at the picture, I stand back and take a look at the whole museum, if you know what I'm saying, so it better look real to me.

Mike: Yes, yes I understand exactly what you're saying. In fact, excuse the pun – but 'I get the picture' [Laughs]. So Andrei, what about the Term Vector Database?

Andrei: OK, let me first explain what term vectors are. I know that there are bulletin boards and newsletters of various search engine optimisers who only have a very vague notion of what they are. Basically, what a term vector associated with a document is, is a list of words, in the document, with some weight. So, I take a document and I'm trying to figure out which are the most important words. The word might be important because I've seen it in the title, I see it a lot in the document and you've marked it as key word and so on. So I'm going to give it some weight. So this way, I can have a number of key words associated with the document and maybe some weight associated with them. So for instance, words which are extremely popular, like say the word 'of', obviously this is going to appear a lot in an English document. But it doesn't matter how many times this word appears in your document, because it's not relevant, I'm not going to give it weight. It's just a very common word. But if you have a word like anaemia, which is not a very common word but it could occur many times in your document. With this word, I'm gonna say, this is an important word in this document. So this way, I characterise the document by a relatively small number of key words and their relative importance. Once you have that, you can use that actively for ranking when somebody does an informational query, we can work out what's the term vector value for that and related pages. Or defensively for discovering certain fake links, when you see a page that seems to be about gardening, pointing to a page about that's about extreme skiing – well from that example – that sounds not quite right! So maybe this link is not an endorsement link.

Mike: So this is basically looking for a continuing link theme?

Andrei: Yeah exactly. It's those things that seem to be going randomly all over the place probably do not carry the kind of endorsement or related information that I want, so I somehow will not use them a lot in my algorithm. I know that there is a lot of talk about themes and theme oriented sites and so on... but the point is that, pages in a site which focus on a single theme, whether I'm actually identifying that or not as a search engine it should be a better site. I mean, obviously it's good to have a site which is focused and the linkage surrounding is good, then it's going to do better in terms of ranking. As opposed to a site which just seems to be about everything.

Mike: So basically what happens is that, you take a page and you give it a term vector, a kind of related number or weighting, and then when I perform a key word search the one with the highest number comes up?

Andrei: Well... [Pauses] well the number of factors involved in ranking is huge. There are things like what we call 'query dependant' factors which are depending on what query you just made and what are called 'query independent' factors which are a sort of the notion of the goodness of the page in general. It's a very complicated algorithm which we are tuning and changing literally on a daily basis. And things evolve as we figure out certain aspects. It's very difficult to look at a single thing in isolation. I would have a hard time, you know, short of you coming to me and saying: "If I change this page, in this certain way would I rank better, or would I rank worse"? well I would have a hard time telling you what would happen until I put it into the whole context. So things are not in isolation and they are changing continually. But what the term vectors do, or at least try to do, is, well you know, you can try to Spam and we work at cross purposes here, but what we are trying to do is get the true 'gist' of a page in a simple machine oriented way. Obviously, if I had a human to look at every single page to determine exactly what a page is about, you know, this one is about cars, this one is about animals that sort of thing…

Mike: So the hardest part of your job, is creating an algorithm which looks at a page in a similar fashion to the way that a human would?

Andrei: Exactly! And that's the main ranking factor which we are aiming for. Eventually for the ranking to view as good, or as close to human judgement. I have to say, at this time we're very far from there, but ultimately, this is what we're striving for. It's like, well I don't know if it's the same for you over there in England, but here we have small towns, and they have little libraries. And there's an old lady that runs it. When you go in and ask for a book, she always knows exactly what you need because she understands what your needs are. She understands what the context is, so you know, a little kid who comes in and says: "I want a book about Italy". So she knows that maybe he has to write report at school about Italy so she knows the type of reference book to give him. Now if I come and ask for a book about Italy then probably what I want to do is travel, so she's gonna give me a travel guide – yes… and my ideal for a search engine is that it should work like this.

Mike: [Laughing] Like a little old librarian!

Andrei: [Laughing] Well she knows exactly what you're looking for – so yes, like that librarian…

Mike: It's a very interesting (and amusing) analogy Andrei. But I understand exactly where you're going with it. Anyway, I know that you have other stuff to do Andrei. But I want to ask you one final thing. Some while ago when I was doing further research, I came across the phrase 'temporal tracking', in the way that search engines measure, what is probably easiest termed as 'click popularity' – is temporal tracking the correct phrase for that type of process?

Andrei: I'm not sure various people use various terms. Some talk about temporal tracking meaning that they have a cookie that is not permanent. It's a cookie that could last temporarily on your machine and then they would track what queries you just made. But then at the end of an hour they would just drop this cookie so that it doesn't stay with you forever. Other people talk about 'click thru' tracking. We do click thru tracking here at Alta Vista but we don't use it for ranking. The reason we don't use it in ranking is that it's very easy to Spam. You just need to write a robot that would search on particular queries and then quote-unquote 'click' on a particular URL over and over again to try and convince me that this is the best result for this particular query.

Mike: Does it work the other way around as well Andrei. In that, if I perform a key word search and then click on the first result, if I don't like what I see and then click 'back' to the results page again, if that happens with this page on so many occasions, then in fact it could be penalised for it.

Andrei: That's correct, that can happen. You know Direct Hit was based on this idea. And they also study how to do 'inferences'. How to do data mining on the click patterns. They look at, like you say, if you click quickly onto a page and only stay on it for five seconds then it's not satisfactory, but stay on another page for 50 seconds then this seems satisfactory. At this point at least, we do click tracking in a way, to validate some of the changes we're making and for Spamming filtration, but we don't use that information for ranking. If I notice that for a particular query, like say, anaemia again, if I see that this page gets clicked a lot, I'm not going to change it's ranking just because of that. Direct Hit would do this, but we do not. So the paradigm there, I guess, is that people vote with their feet as in number of clicks. And the paradigm used at most engines, including us, that people vote with their links. So a page that has lots of links, that's a measure of popularity. The click thing may work, but very often it's a case of how good your abstract is, or how fancy the graphics are, you know, they may have a fancy site, but it may not be appropriate. So you have to be very careful how you use this kind of click thru information.

Mike: This has been so interesting Andrei. You have no idea how much I've picked from just this conversation. It really is so useful. Thanks so much for giving me the time.

Andrei: Mike you're more than welcome. I've very much enjoyed talking to you also. Any time again. Thanks.

Brian Pinkerton

As I've already mentioned, Brian has earned his place in Internet history. In this interview, he explains how he developed the web's first full text retrieval search engine and went on to become VP at Excite. Most importantly, he clears up the confusion that many SEO's have had about the Term Vector Database and explains how he used the 'vector space model' at the very beginning with WebCrawler

Mike Grehan talks to Brian Pinkerton.

Mike: Brian, thanks very much giving me some of your time.

Brian: Sure Mike no problem, go ahead.

Mike: In my time honoured fashion, let's do the background first. Web Crawler was the web's first full text retrieval search engine. So how did the project develop and what was the inspiration behind it?

Brian: Mmm. It, well it was sort of an accident actually [Laughs]. At the time I was a graduate student in computer science in molecular biotechnology. The web had just really started to come into its own and people were really getting interested. I mean it had been around for a while, but it was just starting to gain momentum and people were trying to figure out at the time what they could do with it. And all my fellow students at the time had plenty of time to surf the web and look for cool stuff. I didn't because I was actually so busy at the time. So I just developed this thing to help me find stuff. Initially it was just an application which ran on my computer and then I was persuaded to make it available for other people on the web itself. So – I did that and that's how Web Crawler was born [Laughs].

Mike: [Laughing] There wasn't really a great deal of stuff to crawl out there at that time though, was there Brian?

Brian [Laughing] Exactly, the database had like, six thousand sites in it, or something like that. It just ran off my desktop PC and well... yeah it was incredibly small when you think of it now.

Mike: But it certainly did help to make your mark in Internet history Brian , that's for sure.

Brian: It did... and who would have known [bursts out laughing]. It was just... well the subsequent year that followed is kind of like just a blur as things happened so quickly. Trying to keep up with how fast the web was growing and how fast traffic on the search engine was growing...

Mike: It seems to me that there was that very, very rapid rise with Web Crawler's popularity. Most of the research that I've done on the development of search engines points back to universities, just like your own. But very quickly they seem to be pulled out of the universities and into the commercial world.

Brian: Right. Exactly, and well, I was kind of lucky in that, I did have a year of pure university time with it before it turned commercial. I think that was something that helped me to keep my perspective. Even though then, you know, I was running a business, I really wanted it to have editorial integrity and not compromise the search results and stuff like that. Whereas, for many of the other competitors it was just a rush for the money.

Mike: So, you were in at a very early stage. As you know, I'm trying to track the technology from the very early days to the present time. There seems to have been many rapid technological changes since Web Crawler debuted. Do you monitor the changes yourself?

Brian: Yeah, most of them yes.

Mike: So, as you know, I'm trying to get a clear definition of the term vector database. I know that you used the vector space model in the early days with Web Crawler. There is obviously a difference between the vector sapce model and the term vector database. It's probably difficult to simplify, but could you try and differentiate the two?

Brian: Yeah, OK. The vector space model is really only a means for determining which documents get returned for a query and then how they rank. And really, without kind of boring you with the details of... why it's called what it is [Laughs] really all the vector space model does is to determine which documents contain the same words which have been used on the query.

Mike: And this is about the proximity, about how close they are. Yes?

Brian: Well yes, it's about how close they are and... well lets say you put in a ten word query – well maybe no documents have all ten words in them. And maybe no documents contain phrase fragments which are in the query. That's what the vector space model has to contend with when there's that 'fuzziness' as we call it. So maybe the top document has nine of the words and then maybe the second document has eight of the words and so forth. Here's just a lot of detail running around the edges when you have that kind of thing when you have lots of documents and lots of users.

Mike: I suppose when we have something like Google talking about having three billion documents indexed, it has to be too difficult to rank according to that kind of model.

Brian: It's just a bit too simple and bit too prone to... what's the word... well abuse I guess...

Mike: This is Spamming you're referring to then?

Brian: Yeah, exactly. So lets flip to the other side. The term vector database which Andrei was talking to you about is, basically a way to create something more like, well a synopsis of what a particular document is all about. And that's useful in a lot of different applications. For example it's useful to figure out if two documents are very similar. If for instance one would be returned in a query, because of the similarity, maybe you also return the other one. Say you had a query which only identified one document out of a billion – well this is not a very satisfactory response so maybe you want to find some other ones which are similar that the query didn't identify precisely because of some small criteria – well maybe you want to expand that query a little bit and find some similar ones. The term vector database would let you do that. I have to say it's not perfect. And from what I understand of the term vector database at Alta Vista is that, it's not used in the core search engine. It's not used in the every day process of answering queries – who knows maybe they have figured out by now how to use it for every day queries – but I think primarily it's used along with the connectivity server in a number of ways. Well, here's how I would use it. I would use it to make answers to some queries better. As I mentioned earlier, if you have a query which doesn't turn up too many good results, you could definitely use the term vector database to improve those results. If you have the ability to have processing ahead of time on the index, it helps you to determine which pages are the most representative. In fact, right down to which pages you want to include in your index at all.

Mike: There are some interesting applications built on top of the term vector database.

Brian: One application of the term vector database is to create a... Well, Imagine that you had a hierarchy, like the one at yahoo! or even Open Directory or whatever. If you have that kind of hierarchy, you can use the term vector database to assign pages into the hierarchy. And that kind of automation makes your editors a lot more useful because they can then vet the automation process and concentrate on keeping the hierarchy as clean and as up to date as they can. Rather than worrying about every individual page assignment.

Mike: Andrei Broder explained the characteristics of search as he saw them and broke it down into three classes: classical information retrieval on the web when you're just searching for any material on a given subject – a navigational search when you really just want to get directly to a specific web site – transactional search when you want to get involved and interactive like shopping or downloading stuff. So if you're working from a hierarchy in the way we are talking about then it's so much easier to determine that classification of the search.

Brian: Yeah you're right. This is one of the things which has so far, kind of eluded search engines. And that's the thing – actually knowing the purpose of the users query. And I think the best way to think about that is to... OK, you have a journalistic background so you probably spend a lot of time in libraries researching...

Mike: [Laughing] I spend all of my time researching subject matter.... Maybe I should have been a spy or something...

Brian: [Laughs] OK – so you're in the library and you say: "I'm going to do.. so and so... and I need information about such and such. And from those two things the librarian can determine something. Like if I say, I'm going on a vacation and I need information on... Bora Bora... from those two things the librarian knows where you're gonna go and what you want to do... but also knows exactly what information you need. And that kind of intuitive knowledge is something which is just not available to search engines. You know, if you type Bora Bora into a search engine it doesn't know whether you want to go shopping for a... Bora Bora 'thingy' – or whether you want to go to Bora Bora – or whether you want to write a term paper on it.

Mike: yes, this is the difficulty, this is why you end up getting five million pages returned on some searches.

Brian: Exactly. And if you actually talk to a librarian, that's their kind of chief complaint about search engines. There's no task specific or contextual information around the query about the kind of results you want and things like that.

Mike: It could be made quite a lot easier if it weren't for the fact that... so many people who use the web and search engines to do searches... they're not experienced searchers. They key in just a few words, or even single words, like for instance they type Madonna because they want to buy the new album. If they were a bit more advanced with the query then they would get more accurate results or tighter results back. But, with due respect, your average surfer is just not sophisticated enough.

Brian: Exactly. That's very true. And also, I would say, you can put some of the blame on the page designers who simply don't make pages that index very well. So you can go searching for things and you know that they're there, but for some reason you can't find them. You know that there is a United Airlines site, but for some reason the designer hasn't put the text on the page that you were looking for.

Mike: Again, this is something that I've come across many times in terms of design. In fact, it's one of the points that I was going to make. However much the technology changes, the emphasis is still very much on the textual content of the page. So even if the new technology makes it easier for a search engine to be more accurate and relevant with the results it returns – it is still very much about the keywords on the page which make it count.

Brian: And that's a real difficulty. Now there's a couple of ways around that. One of the very first ways around that was the use of meta tags. You know about meta tags...

Mike: Brian, most of the people I speak to about search engines immediately throw in: "Got to get your meta tags right!" As if meta tags were the answer to everything on the internet...

Brian: Yeah, I know what you mean, but in a purely cooperative way they can be very useful. But there are two problems with them. Two chief problems. One is that they are incredibly useful for Spammers. You just can't trust people to make their own meta tags because they just make stuff up. And the other is that, you can't trust people to make good meta tags either. Even people who are highly trained editors come up with crappy meta data for their stuff [Laughs] it's really amazing!

Mike: [Laughing] I have to admit that, I always look into the source code on most sites I visit and take a peek at the meta tags and I'm constantly astonished at what I see. They can start with 'automotive parts' as a key word and end with 'britney spears' or something and I think: "Wha"!

Brian: True, exactly. You know, I've done some consulting for companies which need search on their own sites, you know, their private collection of editorial data. And they have people writing good stories yet they can't come up with the editorial bandwidth to tag those stories correctly. These are smart people – they're writers for a living and they can still mess it up. So if the people who ought to be able to do it [write good meta tags] can't...

Mike: I have the same problems myself at times [Laughs] Actually, that brings me to the next point. This term vector database thing has had the whole of the webmaster community talking about themed web sites. You know, the thought is that every web site has to have a theme and if you can't sum up your web site in two or three words then... you may as well pack up and go home kind of thing. Is that really the case these days do you think?

Brian: It's not necessarily the case at this time, but I think it's inevitable that as a consequence because of the size of the web...

Mike: It's easier to focus a page on a specific topic, yeah?

Brian: It's more about the link structure of the web I think. In fact, this is what one of my biggest fears is, with all of this sort of stuff, is that we're gonna head down that road and end up with some sort of 'gate keepers' on the web that make some sites successful and other sites not so successful just by virtue of the fact that some other sites point to them.

Mike: Well true, this is one of the other things with link popularity and the connectivity database and all this stuff, where they're looking at a page's actual 'reputation' as it's seen by everybody else on the Internet i.e. the number of people that point back to a particular page. It's not just the number of links, it's the quality of the links, the connection of the subject matter. And it must be very nice if you're a big well known resource site and everybody is happy to point back to you. But if you're not… where do you get these people to point back to you in the first place – I mean how do they find you to want to point back anyway – not in a search engine, because you don't have any link popularity or reputation!

Brian: Exactly! One of the biggest problems with that method is that it doesn't take into account emerging resources. I mean, if you're the brand new thing on the web – well of course you're not going to have any links pointing back to you [Laughs] you might be the cat's meow as far as… well… content on Bora Bora goes…

Mike: [Bursts out laughing] But if you can't be found on a [expletive deleted] search engine… how does anyone know you're there?

Brian: Yep! That's the thing and it's a real problem with the link method. There's various methods for dealing with that and one of them is just to make a point of noticing when new stuff comes up [in the database] and then, sort of, serendipitously include some of the new stuff in your results. That's one way. And there's other ways of mining the connectivity database. You could make an editorial distinction that you're not always going to include the most popular stuff right at the top, you know like super general research. I mean if Google didn't do something – Yahoo! would be at the top of the results every single time! [Laughs] right!

Mike: And you do get this kind of temporal thing which I noticed you cover in your thesis. This is a bit of a confusing thing these references on the web to temporal tracking. I mean, if somebody does a search on Osama Bin Laden, then you're gonna get CNN or whatever the most current page is at that time. but then, this page is not going to be the most popular page (in terms of rank) in six months time or more is it?

Brian: That's right. Exactly. It's a really difficult problem. One of the ways in which search engines achieve… well I know some search engines do this, but I don't know about all of them, but they do actually editorially determine the results for the top, say, thousand queries.

Mike: [Surprised] Really?

Brian: Yeah, so somebody will sit down and say, you know the query… travel… is very popular… and you know, if you run a query like 'travel' on any search engine (even Google) the results are essentially meaningless. So why not have an editor just whack out 25 good travel sites and stick 'em at the top!

Mike[Laughing] With a bit of lateral thinking that's the answer… forget the technology, just roll your sleeves up and get the job done! Strangely enough, I mentioned something like this kind of scenario to Craig Silverstein at Google. And I did say to him, you know, if at the end of the day if somebody has published a site and it has got great content but they guy doesn't know anything about the whole search engine optimisation thing, then the guy really doesn't stand much of chance, so what about if he just emails you and says: "take a look at my site it's great but you're not getting it in your results. When you do a keyword search at Google on my subject you return a lot of crap. So why don't you just stick my site at the top". And Craig did honestly say that if they were missing a great site from their results then they probably would give something like that a bit of consideration. However, it would have to be a pretty authoritative site!

Brian: Well exactly, all search engines have various strategies for dealing with things like that. Unfortunately, the thing that you've just described is pretty much a rare case. Most frequently what happens is, somebody sends you an email that says my site is the world authority on so and so stick it at the top of the results and when you check it's usually anything but the world authority on anything! You know, it's something that's only vaguely about the subject most of the time [Laughs]

Mike: Yes, they'll try anything when Spamming doesn't work. Actually, on the subject of Spamming, I was talking to Ralph Tegmteier who is a search engine optimiser based in Europe (I have to say he doesn't have many kind things to say about search engines) he's a 'cloaking' specialist and I've noticed when I mentioned the word 'cloaking' to a search engine' it's like I swore at them or something. What are your thoughts on 'cloaking'?

Brian: It's a huge problem. You know I understand the desire for the search results to reflect honestly what's on the web. That should be everybody's goal it is the service that they're trying to provide to the users. The search engines do try all the time to really reflect what is out there. I would say that, in the early days of the web, people did a reasonably good job of... well... how should I say it...

Mike: They were a little more ethical perhaps...

Brian: They were certainly more ethical, they were interested in making sure that their stuff could be found... but not overly so, if you know what I mean...

Mike: Yeah, yes I do.

Brian: In the early days of the web it was basically all text. There were few pictures, there was no JavaScript there was no Java, no Flash no frames, none of this junk really. So it was a much easier process...

Mike: Yet it's this junk that you call which causes problems with search results and this is why Ralph justifies using cloaking so that he shows the search engine an optimised page and the surfer the real deal. Providing both things are the same topic i.e. the surfer is getting to see material related to the keyword search, then he believes that this is fair.

Brian: Yeah well I think that, if you could ensure that really is the case – then that's fine. The problem is that the people who try and do that in a legitimate way get hosed by the guys who are just using it to Spam.

Mike: So we're back to the simpler version of the same thing which was using a redirect. You know you do a keyword search for Disney 'toon characters and 3 seconds later you've arrived at Madame Lazonga's Massage Parlour or something.

Brian: Yeah true and that really is the problem. If there were people doing cloaking that you could rely on, people doing the right thing, that would be the greatest. But most are just trying to create traffic drivers. Now that there's been a little bit of a dot com meltdown, there may be fewer people doing it.

Mike: It brings me to the next subject Brian with you mentioning the dot com melt down. There have been many changes since WebCrawler started in the search engine sector. Disney bought Infoseek, re branded it and then just as quickly dropped it. NBC bought Snap, re branded it and then did the same thing. There are rumours that Alta Vista is going broke (although I have the official word from them that they're not) and @Home have jdropped Excite [at the time of the interview this was a rumour. Since then Excite has been sold to Infospace for $10million after an original purchase of $7.8 billion]. What are your thoughts about the state of the industry?

Brian: Well I think... well you know I was a participant in all of this, although I have to say that I was an unwilling participant in all the hype. I always felt that, what happened in the time period from mid 97 to say, middle of last year I always felt that, well I thought it was all just garbage.

Mike: [Laughing] Well yeah... I guess everybody realises now after they've seen all of the dot bombs as they are...

Brian: I think that search is a great small business, right? And I think Google are proving this, you know they're focused on search they're having a small successful business and they're proud of it. The problem is they've had some venture money and maybe they'll have to go public to satisfy their investors.

Mike: Is it very difficult for search engines to come up with a good business model to create a decent income stream. I mean you were VP at Excite.

Brian: I think there are two difficulties here. One is when your customers aren't your users you got a big problem. The people who pay you money are the advertisers. The people who use your product are the surfers. And their interests don't line up unfortunately. One wants to exploit the other and the other doesn't want to be exploited. I don't think that's a good recipe for a business. It's possibly an OK recipe for a small business as long as you got some integrity. And so far, Google appears to have enough integrity to not exploit their users.

Mike: Although they have just gone on record as saying that they are looking at pay for inclusion schemes like those with Inktomi and Alta Vista...

Brian: That's where the money is unfortunately.

Mike: What I tend to find, and I'm sure you'd agree with this in view of what you've just said, is that, I look at these PPC engines and think to myself: I've just performed a keyword search and the results I'm looking at are not the most relevant documents on the web, this is not the best of the web, this is what someone is paying for me to see!

Brian: Right. It's kind of like the Yellow Pages model here in the US where you flip to the automobiles page and the people who paid the most money have the biggest ad. So Avis pay for a one pager, Hertz Rentacar is over there and Fred's Rent-a-car is down there in small text. And that's a fine model as long as the users know what they're getting. As long as they're personally happy to wade their way through those kind of results.

Mike: In the knowledge that they're not actually searching through a series of results from the most relevant documents on the web. They're actually perusing a series of adverts in effect.

Brian: Personally, I believe that the best search engine model is a small business, maybe even a non profit business, you know like consumer reports of the web, or the library of the web, something like that. Maybe even a kind of business where the searchers pay the search engines (however much they would have to pay I don't know).

Mike: Sure, just like the nominal sum you pay for your library card...

Brian: Yeah, exactly. I don't know, maybe you pay five bucks per year just to erase the ads from the site and stick to the real results. I'd pay that. I think with a lot more thought to the model there is a business for that out there. You know this whole search engine as a portal thing, as a media company is just a distraction.

Mike: Is it something you would do again Brian? I mean would you get involved again in this business?

Brian: I would certainly get involved in search again. But really helping people find stuff. Yeah, definitely. I'm really interested in the problems that go with it. You know, even with the success that Google's had, I still don't think we've seen the last word on what constitutes good search on the web.

Mike: So. Which way do you think the technology will go?

Brian: I still think that there's still room for searching the invisible web as it's known. I know we touched on this earlier, you know, the stuff we talked about before when we talked about cloaking and that stuff, you know things which are not currently searched by search engines. So, for example if you're searching for discussions on a particular topic, well all over the web there are forums and that kind of thing, like Usenet is included on Google for instance, but you really can't find where the action is on say, bicycle technology without still spending a little bit of your own time doing it. So it would be great to have something more targeted to help you find where the discussions are. It would be great to find things like, I don't know, maybe government documents on a particular topic are, it would be great to wrap all that stuff into one ball that decides which thing to search first.

Mike: Is it likely to go that way so that you end up with really tightly themed search engines, like tight vertical search, so that if you want to find stuff on, I don't know, maybe extreme skiing (Andrei Broder mentioned that [Laughs]), or just a completely sports related search engine and if you want stuff on, say chemistry you just go to the chemistry search engine...

Brian: I think that's a reality already, I think there are already tightly themed search engines. In fact, just the other day I was, well let me tell you first, I do triathlons, so I was looking for stuff on how to get faster and I actually found a triathlon only search engine! I would never have known that existed!

Mike: Rather you than me – the triathlon – now that... well you must be living a very healthy life Brian, that's all I can say!

Brian: That's right. But the interesting thing about the triathlon thing is that, it's a focused search engine, but it's still a web search engine. It doesn't search the invisible web, it doesn't search other stuff. So, I think it's getting to the point where you need people to make these kind of focused search engines, but the technology for this really needs to be worked on.

Mike: I have to mention Google here again, I did a search at the beginning of this week when I was continuing my research and, of course, I know that they're indexing .pdf documents, but I actually discovered that they are searching Flash now, not for indexing, but checking for the links that it may contain. I figured that out myself. Then I checked for some different file types on an advanced search, and sure enough there were MS Word .doc and .xls and .ps

Brian: Yeah, I think that sort of thing is inevitable, I mean anything that your average person is capable of seeing with one click should be indexed. And the process of looking inside Flash and JavaScript and Java for the links is almost a requirement. I mean so many of the links are hidden inside that kind of thing. In fact, we used to go as far as to monitor the news-feeds for links. So we would take Usenet news-feed, which is 50Gyg per day and just suck the links out of it! We stick them in the search engine on the theory that, well there were a couple of news-feeds at the time where people posted details about the fact that had a new site about one thing or another. So the way then to find put about something new was in the links in this news-group. And so using other media sources as a means of finding new URL's was an obvious way to go.

Mike: Brian, this conversation has been so illuminating and I really do appreciate it (considering with the time difference it's very early in the morning for you). So, what about yourself personally as a final thing. What is it you see yourself doing now and in the future?

Brian: Well, I'm looking around at starting the next thing and I'm trying to decide right now whether that will be a search related thing or to move in another direction. I've got a lot of really strange interests. So I may want to go and look at something new. Part of me thinks that, you know, because I am pretty good at this search stuff that I should go back and work on it, but part of me thinks… well… that was a chapter, now it may be time for a new topic [Laughs].

Mike: Brian, Web Crawler was a great innovation, so whatever you do I wish you the best of luck you deserve it. And if you get back into search that would be great.

Brian: Thanks Mike. So how's the book coming along?

Mike: Well – it started off as just an overview, or the guide as it is. It's now turned into something like War and Peace [bursts out laughing] but thanks for asking!

Brian: Well, best of luck to you too Mike and take care.

Craig Silverstein

Craig is Director of Technology at Google. He's been with Google since before it went commercial and was still a university project at Stanford. Google is, perhaps, the most important search engine online from a search engine optimisation point of view. In this interview, Craig explains the meteoric rise of Google on the web and what Google does and doesn't like when it comes to search engine optimisation. Cloaking at Google? No way. Rank checks every day? No way. And very kindly, Craig also explains how to create a Google friendly web page.

Mike Grehan talks to Craig Silverstein.

Mike: Craig, first of all, tell me what your current position is with Google and a little bit about your background:

Craig: OK, I'm Director of Technology, which means that I'm definitely on the technical side of the business. I look at the projects that are going on and make sure that they're OK. Basically, I've been with the company since the beginning so I kind of have a feel for the company as whole and how it interrelates, so my job is to take advantage of that knowledge. In fact I was with Google before the beginning - I was involved when it was at Stanford and it was still a research project. Larry and Sergey, the founders were still working on it as a research project. I was in the same research group and kind of got interested so I just joined in. And then when they decided to become a company they just invited me along. I thought it was a great opportunity to be there at the start up. To start at the ground floor with such a high quality product in such an interesting field.

Mike: Google had such a meteoric rise from the 'ground floor' as you put it. Seemingly out of nowhere Google has become the number one search engine on the internet:

Craig: Yeah, well frankly, you know, I'm a technology person and when I joined I knew that Google had a better search technology than other search engines out there at that time. But, you know, one of the things you notice if you follow technology is that technological prowess isn't necessarily an indicator of success. To some extent we're lucky that people have embraced Google. I think, however, that the search engine industry is different to a lot of other technology industries and the cost of changing is so low, that if you don't like one search engine it's so easy to change to another. It's not like having to learn a whole new word processing system or something like that, so we think that in our field, actually, quality does win out to some other parts of the technology business. So we're working very hard to make sure that we continue to have the best search.

Mike: Google has won awards as the best search engine in the world surveys conducted by Danny Sullivan. Do you know what the criteria is?

Craig: Well - my understanding of how this works is that, he has people vote for them. I guess people read his newsletter, people who attend his conferences - these are mostly webmasters - and the criteria that they used to judge a winner was which search engine they liked using themselves. And which search engines they believe do well by webmasters. I guess that means if you go to the trouble of making a high quality site, the search engines recognise this and give a high result for the type of queries that are appropriate for them. And we actually come first in both of those categories. I think that webmasters are the toughest audience to win over because they know a lot about the various search engines, they know a lot about the field and so, to have their endorsement is something that we're very proud of.

Mike: We use the term search engines generically for any search service we use on the web. But there is a distinct difference between the two. Yahoo! is the world's biggest directory and Google is the world's biggest search engine: How would you best describe the difference between the two?

Craig: Well, there are two different ways of cataloguing information on the web: One is what you would call 'human powered', the other is what you call 'machine powered'. Directories like Yahoo! are human powered, they actually have human editors which visit web sites and see if they meet the quality criteria they have for making up their index. And then they add it to the appropriate category. Yahoo's index is made up of a number of different categories, like Arts, Humanities, Business and so forth, and this is all done by people. And you have the advantage that, people are much better judges of content than machines are because they understand what they are seeing and reading, whereas machines don't. Yet the disadvantage is that it's very labour intensive. Yahoo can only work on a very small percentage of web pages that are out there. Open Directory is compiled in a similar way, but even they only have a small percentage.

Mike: So how many queries does Google handle per day?

Craig: Over 150 million searches a day, about half of which are on google.com and half of which are on our partner sites like Yahoo!, Netscape, and Cisco.

Mike: It's interesting that you mention Yahoo! and Open Directory together because Google has a link with both. Google supplies the web page results at Yahoo! and Google's directory results are supplied by Open Directory.

Craig: Well it's a little bit complicated. Yahoo!, I think very wisely, combines it's results with a search engine... [at this point, for some reason, Craig drops the subject and moves onto how search engines work]. So, the search engines are the ones that have machines to add pages to their index. They basically go out and look at all the web pages out there. We look at about 3 billion url's I believe, and then we store information about them in our computers. And then when someone does a search, the computer goes through all these web pages and tries to decide which ones are the most appropriate for that search. There's a lot of info technical variation which the computer has to judge to decide how good a web page is and how well it fits into the search query and there's a lot of room for improvement there. Google has some technology of its own that it developed which has turned out to be pretty successful.

Mike: Let's just go back to directories again for a second. Is it always the case, do you think, that if I'm Abacus Communications, as opposed to Zetec Communications, I'll always do better in a search?

Craig: (laughs) Yeah, like the Yellow pages thing. I was once trying to ship my car from Florida to California, and I looked in the Yellow pages to find a car shipping company. I think the top four entries were something like: AAAA Shipping; AAA shipping; AA shipping and A shipping which was at the top.

Mike: Yeah. So which one do you choose? The one with the most A's or the one with the snappiest name!

Craig: So yeah, Yahoo! does have this alphabetical listing which is perhaps not the most appropriate for the web. Google has its own directory offering which you mentioned earlier. It's based on the Open Directory, we don't actually categorise the pages. So what we do is, we order the web pages within a category by our own PageRank methodology which is as much about the quality of the page and therefore we have, sort of, the web's consensus about the best automotive shipping page to be first in the results, other than what comes first alphabetically.

Mike: It's an interesting subject. I mean, say for instance you did a search on Britney Spears, which I think recently came top of a poll as the most searched for term on the web for that particular age group. If my daughter (who is a big Britney fan) put together her own unofficial Britney Spears fan site, I guess if Yahoo were to have a look at that, it wouldn't rank very highly against the official Britney Spears fan club site. However, a search engine like Google wouldn't be able to tell the difference really would it?

Craig: Well, that's an interesting point. A directory, as I say, like Yahoo! which I think does it in alphabetical order (correct me if I'm wrong) that would be the category. In which case, whatever name they gave, that would be where it turned up in the Britney Spears category. For most search engines, you're right, they would have a hard time with that. For Google, one of the reasons; one of the things which sets it apart is this PageRank technology which looks at how web pages come together and it would notice that lots and lots of people link to the Britney Spears official site and that, perhaps, fewer people link to your daughters web site.

Mike: (laughing) What a shame for my daughter - she'll never be found! Actually, it is a point that I was going to come on to. In terms of the technology used, a webmaster of a smaller company, possibly setting up a site on their own, for a while it was always about meta tags. In fact, many still believe if you get your meta tags right and you'll do all right in search engines. But all the more it seems to be shifting towards how popular your site is as a determining factor towards how a search engine would rank your site. Is that a fact, I mean I guess that IBM has so many links pointing back to them than the average web site that they don't ever have to worry about search engine positioning?

Craig: Well, for certain queries, certainly if you search for IBM, then yeah, their home page is bound to come up first. But there are other queries that would be searched on for IBM. Hard drives for instance. They may have to worry there about other companies in the hard drive business. I mean, there it would be a case of: which of these companies is, basically, most popular on the web? Which high quality web sites out their link to IBM when they think about hard drives, so, you know, big companies count as much as small ones. To go back to the Britney Spears example. I urge you to try on Google and you'll find that the official home page is right up there at the top. But there are "unofficial - official" Britney Spears web pages which are quite good pages and sources of information and quite comprehensive which are being maintained by fans. The fact that they're such good sources means that a lot of people link to them also so they're right up there at the tops of search engines. Perhaps your daughter could generate a site like that.

Mike: (laughing) Yeah: And I'd be the one who gets to put his banner ad on it! Anyway. The search engine industry. I've been watching it for a while, it seemed to start as fairly kind of random sort of thing and then it came in to its own right. You hear now about people speaking of 1500 or even 2500 search engines on the web. But at the end of the day, I think it's a fact that over 90% of all traffic on the web comes from between 9 to 11 major search services. Is the industry shrinking, growing or what?

Craig: (long pause) My feeling, and it's just my take, is that it's consolidating. I think with any industry, this happens as it matures. I mean - the internet is still very, very young. But I think a lot of people got involved in it at the beginning and there just wasn't enough revenue to support all that. Search engines do need to make money to survive and typically exist through advertising and through licensing i.e. through having companies use their search engine services, you know to add search to their company web site and that sort of thing. It's tough to have a lot of people going after the same markets.

Mike: Just staying with that subject for a moment. I noticed that, Google will provides their technology free of charge to universities around the world. Is that a bit like giving away the Coca Cola recipe or not?

Craig: (laughing) Well, I don't know about it being the Coca Cola recipe, but more of building a loyal user base from the beginning. College students are going to be searching the web probably for the rest of their lives, depending on the job they do. If we can get them to appreciate the value of quality search early on - then that's good for us .For me personally, coming from a university background and having used university web sites, I know how poor they can be and we're happy to do a, kind of, public service and help them out to be able to provide a better search.

Mike: It has been said that Google tends to prefer to return a .edu result, like a university or higher education establishment. Even a .gov (Government site) in a search query. Is that fact or fiction?

Craig: I guess there are various issues here. We do have a government search, we call it a site search, where we restrict the search to just government documents and we also have one for the various universities. As for whether our generic searches all tend to have an academic bias (long pause) I think that there probably is some of that. And we don't necessarily think that it's a bad thing. We think that often, web pages associated with universities are more likely to be (unlike commercial web sites) unbiased and being willing to link to lots of people or lots of different sources to give out information. So if those sites turn out to be the most relevant for a query - we're not ashamed of that at all.

Mike: You mentioned, in terms of making money that, the industry is consolidating, but I've noticed, like a lot of other people that since GoTo came around, or Overture as it now is, there has been much more of a shift towards 'pay for placement' - bidding or paying for key words - and other profit making tactics. We have Inktomi providing a service for their spider to come around every 48 hours if we're happy to pay for it. Is this sort of move something that Google would consider?

Craig: Google doesn't really have plan to offer these things at this time. Our goal is to have the entire web in our databases. And our goal is also to find web sites that change very frequently and are called on more frequently than any else in order to keep the data fresh. And we think that's part of our job. We don't think that people should have to pay to have us do that, to use our services. We're not necessarily there yet, we've got a good chunk of the web, but not all of it.

Mike: It is a major task that you have ahead of you, looking at the exponential way the web is growing every day.

Craig: So you can understand why as an interim solution companies may want to go for those paid for services. But we prefer to concentrate on the goal that we have of actually being complete and up to date, in terms of time, without the need to do those kind of things.

Mike: Would you agree that it's a case that you may not get a true result of what's out there, in terms of a site's popularity, if someone is paying to be at the top. It's not really as fair as if you were in there with everybody else.

Craig: Well I think, I mean, I'm not quite sure, what answers people hope to get out of these services, but I think that the big advantage of being crawled every 48 hours is that, if your site changes very frequently, the search engines will notice. Search engines don't notice changes instantaneously. They have to go out onto the web to find if a page has changed. And for sites that change a lot, being crawled every 48 hours is to your advantage.

Mike: On the subject of webmasters, obviously there are people who make a living out of search engine positioning by guaranteeing top ten positions. There's software packages which guarantee to get you into thousands of search engines, online services providing similar things and hundreds of e-books and news letters on the subject. Do you ever check out any of these to find out what web masters are suggesting and promoting?

Craig: I have to admit that I haven't read any of the books, but certainly it's very interesting to attend the search engine strategies conferences to see the kind of 'cottage industry' which has grown around the subject.

Mike: Is it likely that search engines will ever have the kind of relationship with search engine positioning companies and online marketing consultants and agencies (quality companies) in the same way that conventional media like press, radio and TV has. I mean could you ever become a recognised, or qualified positioner in the eyes of the search engines?

Craig: It's certainly the case that, there's a range of search engine positioning companies that emphasise having quality content, that other sites are linked to you appropriately and in a way that makes it easy for both search engines and people to know what's going on. We have no problem with that at all, with those companies that achieve better ranking by encouraging good web design. But the kind of companies that try to put, you know, invisible text at the bottom of every page to try and fool search engines. And those which set up their servers to show different content when search engines come and they see one thing, but regular people see another. Those we frown on much more, much stronger. So I don't know that we'll ever come to a point where there'll be a seal of approval or anything like that.

Mike: What about the subject of positioning software. There's quite a lot out there. Web Position Gold probably being the most popular: Is that a 'friend or a foe' to search engines?

Craig: The main problem that we have with products like that is that they do a lot of automated queries. We really like people to see our search results, it helps us to go through our logs to find out what's working and what's not, for search engines to improve the quality of results. And it's also a case that we serve adverts and we want people to see those ad's and be able to give accurate figures to our advertisers. Some online services do a lot of automated queries and we're not so fond of them doing that. However, we do understand that people do want to see how well they're doing in the search engines and we have to be sympathetic to that. But it's also possible to abuse these products and, of course, we're a little less sympathetic to that.

Mike: I was thinking, if there is a shift more towards paid for inclusion like the Inktomi style service, then you'll need to visit search engines with your credit card details. I can't imagine anyone being too happy about a piece of software openly zooming around the internet with your credit card details attached to it.

Craig: It's an interesting point. But as I say, I'd rather that people just didn't use these automated systems anyway.

Mike: I have to ask you this Craig: If I wanted to build the perfect site for Google to achieve a decent rank what would I have to consider in my code and design etc. I know you can't give away too many secrets (laughs) but what advice can you give.

Craig: Actually I'm glad to tell you Mike. The advice that we give people and it's - free advice - is that, there are two components to making a web page that will do well, not just on Google, but on any search engine. One is to have content which reflects the site. You should have text, especially on your home page which explains what the site is about, explains what service or product you're offering and provides easy navigational elements to get to the various information and products you have. Things like requiring people to sign in or having passwords is going to hurt. The less infrastructure you have to let people get to your information the better time search engines will have with it and the better people will also. The other part is linking. It's important that. relevant pages on the web that should link to you - do link to you. Things like directories, Yahoo! and Open Directory. But also including other sites which are related to your area, or field of business. You know, if you're selling Schulz memorabilia then you should have people who have Peanuts pages linked to yours.

Mike: So you're a secret Peanuts fan then! What about 'doorway pages', 'hallway pages' or 'hook pages' as they're variously known. Search engine positioning companies and consultants still swear by them. How would they work better than just having a great content site?

Craig: Yeah I'm not... well there are people who believe they can outsmart search engines by constructing these convoluted link structures that are kind of intended to fool the search engines into thinking that they're more popular than they really are. I don't believe that those are very successful. We don't have any evidence that they work very well. And if a person gets caught up in that it can be very confusing anyway, and be very unproductive. So given the fact that I don't think that these things really help and they're intended to deceive we don't really like them.

Mike: There is a subject that you touched on briefly earlier, the subject of cloaked html or IP delivery as it's also known. The method of using specialist software for server side delivery where search engines are served with one page and entirely different page is served for the visitor. What's your opinion on that - is that something that's frowned upon as you say - or is it a good thing?

Craig: We find it bad in all cases! There is, perhaps a useful purpose for it, for instance, if you want to show a movie to your audience you would deliver that to them, but you may show a text page to a search engine because they don't understand movies. That's the closest, or most friendly kind of interpretation of it. But even that, we don't think is a very good thing. We think that you should put up your text based page as well which may be more helpful to them than the movie if their browser doesn't support that kind of technology. The worst way that it's used is when it's used to show random content to a search engine and something entirely different to people. Like putting the whole of the entire English dictionary or something so they match every query. (Laughing) Now that - we certainly do frown on!

Mike: (Laughing) What a very dull site that would be! So what's the advice here then - don't do it or you'll be penalised?

Craig: You could be 'black-holed' by all search engines completely without a chance for parole if you do cloaking. Search engines depend on knowing what their users will see when they go to a site. Cloaking for even the most innocuous reasons defeats that purpose.

Mike: Is it a fact that search engine spiders can't extract information from web pages using frames and tables? And if so what's the problem?

Craig: No, that's not true. Google deals with both of these just fine, and I'm sure other search engines do too.

Mike: As link popularity is a major factor in Google search, is it possible to use Google to check how many links you have pointing back to you?

Craig: Google does allow you to see who is pointing to you. If you type in a url into Google's search box, you'll end up at an "info" page that offers you several options, one of which is to see what web pages point to you. You can also type in a query like link: www.yourcompany.com directly into the search box, without going through the "info" page.

Mike: I've heard that Google will now search the 'hidden web' as it's known i.e. things like pdf documents

Craig: Yes, we do this already.

Mike: And will this be an alternative search facility i.e. will there be one search for html docs and another for pdf?

Craig: The way we implement this now is we combine html and pdf documents into the results we give.

Mike: Does the Google spider retrieve information from meta tags?

Craig: I'm afraid I can't get into details about this, but I can give a general rule of thumb: Google tries to look at a page the same way a person would, so page elements that jump out to a user (like bold text or titles) are also likely to jump out to our spider.

Mike: You touched on multimedia technology like movies and Flash. This type of technology, particularly Flash, which is becoming more and more predominant on the web. Yet it really does cause problems with a search engine spider because they don't see anything other than a graphic.

Craig: Right - search engines search for text. html has capability to provide where, it's associated typically with images, where if you choose not to load images in your browser, or your browser doesn't support images, or Flash, you can show a text alternative. And quality sites pay attention to this. Search engines look at alt text to get an idea of what they're missing. But yes it is a problem and I think it's the search engines problem to find an answer to that. I don't know, maybe listening to the Flash presentation if it has a voice track and doing speech to text, or something like that.

Mike: Then maybe Macromedia should be talking to the search engines.

Craig: Yeah - that would certainly make our lives easier!

Mike: Talking of technology - that's your background - people still do complain about not being able to find what they're looking for in search engines. Google's won awards, and no doubt well deserved, for its results on key word searches etc. But do people tend to do a single key word search, or all the more do they attempt to do the Ask Jeeves type search where they use a phrase or actually ask a question. Is it more about key words, or key phrases. Is it better on Google to just use a key word, or should it be a combination of words or a complete question?

Craig: So - you're talking about the difference between a key word search and a natural language type question? Google was written to be a keyword based search engine and we find that for keyword type searches we do very well. We also find that for natural language type queries we do quite well. We've put in some support for that and we're doing more work on that, always trying to do better in that area. In fact for many queries we can do better than search engines like Ask Jeeves which are developed from the 'ground up' to be natural language.

Mike: Is it likely that there'll ever be a situation where search engines will be able to assist with a search, for instance if I typed something like 'recipes' I would get some kind of prompt like - do you want meat recipes or vegetarian. That sort of thing.

Craig: There's always work on this. And this is a hot area in research - allowing such query refinement. We've found, however, that users don't really like that. If you offer all these 'fancy features' the users don't tend to use them because it requires more commitment or more time. They like the basic features. My belief is that they're not confident that, at the end of this longer process they'll actually get what they want. I think they would prefer to define the search themselves i.e. only vegetarian recipes to get the kind of results they're looking for. So... I don't know. I think that may take time.

Mike: What about Geographical searches? I concentrate on 'spreading the good word' as it were, in the region I live in. So what if I specifically wanted to find somebody making 'blue widgets' in the north east of England. Is there ever likely to be a situation where Google can give me results just of the 'blue widget' makers in that region?

Craig: Well, certainly we believe that, a perfect search engine would know where you are and would realise that your query was a geographical one. Like a query for a popular theatre in your region and not something across the other side of the world. It would be able to figure out which web pages refer to your part of the world and then return those results. There's also a lot of work going into that. At Google we're already doing a lot of work on geo-targeting, when people are using that as one of the factors and we try to return more relevant results.

Mike: Can I just go back to the link popularity thing with Google. I have been looking at a product which is being promoted on the web, it's called the Zeus robot. And, as I'm sure you're aware, the basic functionality of the thing is that, it will zip around the web much like a search engine spider, but dropping e-mail messages to specific sites in specific categories simply saying: I do this - you do the same - let's link together. I mean it's a lot more sophisticated and polite, but basically that's it. Some people are very much worried that it is just a 'posh' Spam. What do you think?

Craig: I must admit I'm not too familiar with the Zeus product, and I wouldn't want to comment on it specifically in any case. What I can do is respond to the Zeus view of why link Spam is bad, with our view. As a general principle, a big reason that link analysis such as PageRank works is that a link is a vote of confidence in a web page. It doesn't necessarily mean "I use this page all the time" or even "I like this page," but it does say, "I think this page is worth looking at." It's true that not all people create links for this reason, but the higher "quality" a site is, the more likely it is very careful about what links it uses (This is why PageRank is a particularly effective method of link analysis, because it takes into account the "quality" of the site the link is coming from). The best types of links are those which web page authors create because they think the readers of their page might be interested in whatever web page is at the other side of that link. Any link created for another purpose - for instance, to earn a reciprocal link from another website, or because a third party mandates those links - reduces the quality and usefulness of links. Our philosophy in a nutshell: We encourage web site authors to try to get people to link to them - but the method we propose is to have the author contact web directories and other sites (presumably in related areas) whose readers might be interested in the author's site. We believe authors who have good content and follow this strategy will have no problem getting a good ranking in search engines, including Google.

Mike: Craig, much as I really would love to sit here a lot longer and talk, I know that you have your presentation to a 'hoard' of inquisitive web masters to make in just a short while, so what about the future? You've had, as I said earlier, this phenomenal rise in such a short time. What's next? What will be the main changes that we'll see with Google in the next year or so?

Craig: (Thoughtful for a moment): Well, it is hard to see that far into the future. The internet industry is so young and changes so quickly. But... I can tell you where we're trying to go as a company (and I don't know how long it will take to get there). Our mission statement is: To make the world's information universally accessible and useful. And there are several components to that which we're continuing to pursue and one is - gather all the world's information. Not just what's on the web, it means getting information that's not on the web. We took a step in that direction by acquiring the archives of Usenet data from Deja. And the other thing is to make it universally accessible and useful. Which means things like you mentioned yourself, like being able to figure out whether a query is for local information and having a better understanding of natural language queries. All of these things will enable search engines to give you information the way people give information. I think that's really the eventual goal.

Mike: Time's run out unfortunately, but that was great Craig. Very informative and useful - just like Google itself!

Craig: No problem Mike. You're welcome. Anytime.

A BRIEF HISTORY OF SEARCH ENGINES

This brief history of search engines, is, just that: brief. A condensed version of where it started. It's fairly safe to say that the history of search engines, as we know them now, starts with university student projects which evolved into major commercial organisations. Prior to the dawning of search engines and directories the world wide web was in a chaotic mess - the biggest librarian's nightmare in the world. There was information, tons of it in fact, but you just simply couldn't easily find it. Even with today's extremely advanced search technology, some people still believe that not a lot has changed. This is not quite true though. Search engines and directories have at least attempted to provide a more methodical and logical method for retrieving information from the billions of pages which exist on the world wide web. And the number is growing exponentially every day. The work which started as university projects has revolutionised methods of information retrieval and the way we use the web today.

Although we tend to use the term 'search engine' generically for any type of search service available on the web, they fall into two distinct categories: search engines and directories. This guide goes into great detail to explain the difference between the two and why, for online marketing purposes, you should never confuse one with the other.

I think it's important to have a basic background knowledge of the rapid growth of search engines and development of the technology.

Around about 1990 and prior to Tim Burners-Lee's introduction of http and the world wide web, Alan Emtage, a student at McGill University in Montreal, Canada, wrote a programme called Archie. The programme was one of the earliest attempts to provide a method of identifying and retrieving files on the Internet. This was followed by Gopher, Telnet, Veronica and even Jughead. There's a wealth of information about these early programmes and their strange names available on the web for anyone who really wants to study the history of search on the Internet, but for the purpose of this report it's not essential information. This brief look at the history of search engines really begins with the World Wide Web Wanderer, the first real robot on the web. Developed to capture urls on the web to measure its growth, it resulted in Wandex, the first web database. And so, robot technology, or spiders as we know them now, became the popular university projects mentioned earlier.

Spiders are computer robots (software programmes) which automatically perform repetitive tasks at speeds that would be impossible for humans to achieve. Frequently referred to as 'bots' the term mainly refers to those which traverse the web looking for html pages to be compiled into large searchable databases.

From the names and functionality of Internet software programmes you may never even have heard of, let's leap forward to a name you will surely have heard before: Excite. Still en extremely popular search engine, Excite dates back to 1993 and started as (yes... you're right!) a project at Stanford University. The project was known initially as Architext (the spider still is) and eventually became known as the Excite search engine.

Let's stay at Stanford University for the moment, but leap ahead just a couple of years to another major milestone in search engine history. Student Jerry Yang and his friend David Filo had a project cum hobby on the go. On their own web site, they had started indexing other web sites which were of interest to them or that they were simply aware of. What started as a hobby with the possibility of making a few dollars now attracts hundreds of millions of users and at this time is still possibly the best known brand on the world wide web - Yahoo! However, the difference between what Yang and Filo were doing, compared to that of the other students who were concentrating on i.e. spider technology forms the fundamental difference between what actually are search engines and what are not. Yahoo does not use spider technology. It is the largest human edited directory on the web. As the directory grew and grew it became necessary to implement some search technology into the database to make it more easily searchable, so in that sense, it does sort of 'blur' the line between search engines and actual human powered directories. However: a directory it is.

As the 'bot' technology continued to develop rapidly, it was becoming easier to locate documents by url, title and even the first hundred words or so on the web page. But (wait for it...) another student project was taking place at the University of Washington. This project, like the rest, started simply as a way of finding and storing information on the world wide web. The difference with this 'bot' though, was its ability to search the full text of entire documents. And so, Webcrawler takes its place in the search engine hall of fame (Webcrawler was bought by Excite in 1997). The race was now on for market share in the search engine industry as WebCrawler was followed by Lycos and Infoseek (Infoseek was bought, renamed as Go network and then dumped by Disney Corporation).

Now to another milestone: Alta Vista. Operational from December 1995, it rapidly began to stand out as the search engine on the web. Fantastic speed for query results and the first to use natural language queries as well as advanced search techniques including Boolean Operators, it stood out from the rest. It was also the first to provide what is now a much needed resource: the ability to check for other url's pointing back to your own site. Link popularity is a very important factor in search engine positioning and is covered in depth later in this report.

Time to visit, yes, another University. This time the University of California. Where Eric Brewer, an assistant professor of computer science at Berkeley, and Paul Gauthier, a graduate student, decided to use their collective knowledge and research for a commercial venture. In 1996 Inktomi Corporation was formed and gave the web Hotbot.

They took the web by storm and rapidly became one of the most popular of search engines. Able to index 10 million pages per day and with its advanced use of 'cookie' technology it became a favourite target for search engine optimisers. Although Inktomi itself may not be among the better known of the online brands, its importance for online marketers is great. Results from Inktomi are pulled in from a number of other online brands and portal sites. It was also the first crawler based service to add a paid for listing service. This report covers the Inktomi operation in full.

What about a search engine which searched other search engines for results? Enter Metacrawler. Developed in 1995 by Eric Selburg, at the University of Washington where WebCrawler was also developed. Metacrawler searched Lycos, AltaVista, Yahoo!, Excite, WebCrawler, and Infoseek at the same time.

Late arrivals, but almost immediate superstars are Larry Page and Sergey Brin. Again, from a project at Stanford University, the two have taken the project in 1988 to the current number one spot on the search engine chart. There's more detail about Google and its importance in the section on 'How Search Engines Work', my interview with Crag Silverstein, and also in the Major Players section of this report.

Since the first edition of this guide, we've lost some and gained some. Gone are what was Infoseek and NBCi which had been Snap. Since the first edition of this guide, we've also seen the demise of Excite, one of the earliest major brands on the web which is now owned by Infospace. Although the brand does actually remain, the results are now aggregated in a meta search.

GoTo which was the first of the 'pay Per Click' search engines has been re-branded as Overture and continues to go from strength to strength. The search service

Direct Hit which powered Ask Jeeves has gone and been replaced by 'new kid on the block' Teoma. While Looksmart directory has bought the other new contender Wisenut.

All of the existing major players are covered in full as well as the search engines new for this second edition of the guide.

It remains to be seen what the playing field will look like come the third edition.