Search Insights 2018

April 2018

The Search Network

Contents

Editorial services provided by Val Skelton (val.skelton@blythespark.co.uk)
Design & Production by Simon Flegg - Hub Graphics (simonflegg@hubgraphics.co.uk)

Introduction

Martin White

In a statement from The White House dated 6 October 1965, President Lyndon Johnson observed:

> *"The fire of progress is lit by inspiration, fueled by information and sustained by hope and hard work. Efficient management of any large-scale enterprise – whether in government or business, science or technology, depends increasingly upon readily accessible sources of information. Modern methods of storing and retrieving information are essential to sound judgment, improved efficiency and lowered costs".*

At the time that President Johnson made this statement significant progress had already been made in the development of the information retrieval technology that is the foundation of current search applications. Jump forward to 2018 and organisations are still struggling to find business-critical information because of under-investment in both search technology and a support team with the appropriate skills.

Many of these organisations are now seeking to enhance or replace their current search applications to take advantage of developments in natural language processing, artificial intelligence and machine learning. With no recent experience of how to select and implement search technology they are usually unaware of the range of search software applications that are available and how best to go about the process of selection, evaluation and implementation.

Search Insights 2018 is a collection of essays by members of The Search Network about how to approach this process, maximising the benefits to the organisation whilst reducing the risks inherent in any novel project. We are all practitioners running our own businesses, so we have to exceed the expectations of our clients more used to working with large software and integration companies. Together we have well over 50 years of experience in helping organisations to find business-critical information, working with enterprise search, e-commerce and web site search, and with specialised search applications.

Our objective in writing this report is to summarise some of the insights we have gained from these projects and make this knowledge open to the search community worldwide. That is why there is no charge for this report, and it carries no sponsorship. Not only do we work with different types of search applications, but we also write in our own style and from our own individual experience.

We would encourage you to make contact with which ever consultant you think could best help you with enhancing the search experience you offer to your employees. Members of The Search Network are located in the USA and Europe and can work across both regions in partnership as a client requires.

Our most significant contribution to our clients is a very good understanding of what an effective search application can deliver in terms of business benefits and employee engagement. Very few organisations have had an opportunity to see and use the range of search applications that we have worked on. We hope you will find that Search Insights enables you to make the right decisions about providing your organisation with effective access to information.

David Hobbs, David Hobbs Consulting **(USA)**
David helps organisations make higher impact digital changes, especially through early strategy to best frame these initiatives before they begin. He is the author of Website Migration Handbook and Website Product Management. His clients include the Center for Internet Security, the Library of Congress, the Mideast Broadcasting Company and the World Bank. Follow David on Twitter @jdavidhobbs.

Charlie Hull, Flax **(UK)**
Charlie is the co-founder of Flax, which builds open source search and Big Data solutions for clients worldwide. He writes and blogs about search topics, runs the London Lucene/Solr Meetup and regularly speaks at, and keynotes, other search events across the world. He co-authored Searching the Enterprise with Professor Udo Kruschwitz. Follow Charlie on Twitter @FlaxSearch.

Miles Kehoe, New Idea Engineering **(USA)**
Miles is founder and president of New Idea Engineering (NIE) which helps organisations evaluate, select, implement, and manage enterprise search technologies. NIE works and partners with most major commercial and open source enterprise search and related technologies. He blogs at Enterprise Search Blog and tweets as @miles_kehoe, @Ask Dr Search and @SearchDev.

Helen Lippell (UK)
Helen is a taxonomy consultant. She works on taxonomy development projects, including taxonomy audits, ontology modelling, tagging initiatives, semantic publishing, metadata training and more. Her clients include the BBC, gov.uk, Financial Times, Time Out, RIBA and the Metropolitan Police. She writes and speaks regularly and is the programme chair of Taxonomy Boot Camp London. Follow Helen on Twitter @octodude.

Agnes Molnar, Search Explained **(Hungary)**
Agnes is the managing consultant and CEO of Search Explained. She specialises in information architecture and enterprise search. She shares her expertise on the Search Explained blog and has written and co-authored several books on SharePoint and Enterprise Search. She speaks at conferences and other professional events around the world. Follow Agnes on Twitter @molnaragnes.

Eric Pugh, OpenSource Connections **(USA)**
Eric is co-founder and CEO of OpenSource Connections where he helps federal, state and commercial organisations develop strategies for embracing open source software. He co-authored Enterprise Solr Search, now in its third edition. He is interested in how Search is being invigorated by Machine Learning and exploring approaches for sharing data the way the open source movement shares code. You can follow him on Twitter at @dep4b

Doug Turnbull, OpenSource Connections **(USA)**
Doug is CTO of OpenSource Connections and the author of Relevant Search. His goal is to empower the world's best search teams. He has assisted with search at organisations in a variety of domains. His clients include Wikipedia, Snagajob, Careerbuilder, and many search organisations. Follow Doug on Twitter @softwaredoug.

Martin White, Intranet Focus Ltd **(UK)**
Martin is an information scientist and the author of Making Search Work and Enterprise Search. He has been involved with optimising search applications since the mid-1970s and has worked on search projects in both Europe and North America. Since 2002 he has been a Visiting Professor at the Information School, University of Sheffield and is currently working on developing new approaches to search evaluation. Follow Martin on Twitter @IntranetFocus.

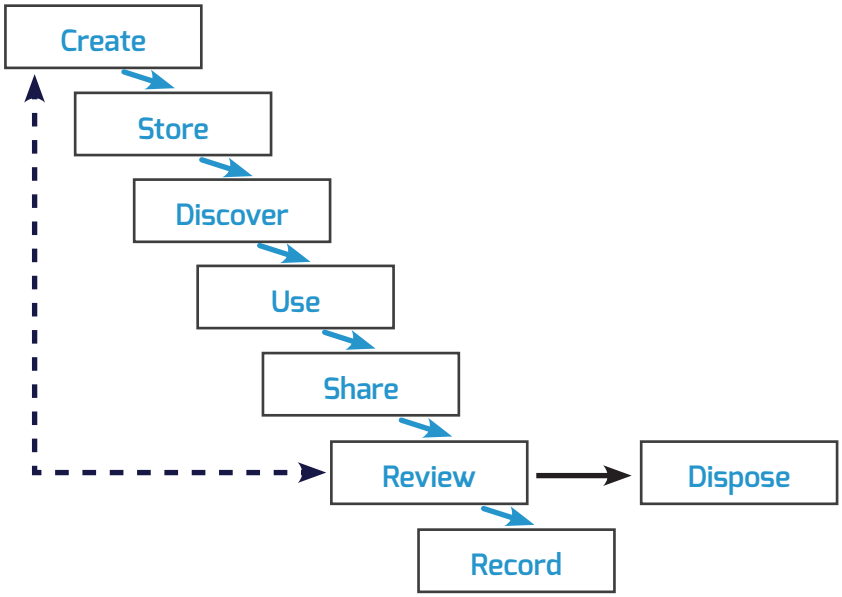
The business case

Martin White

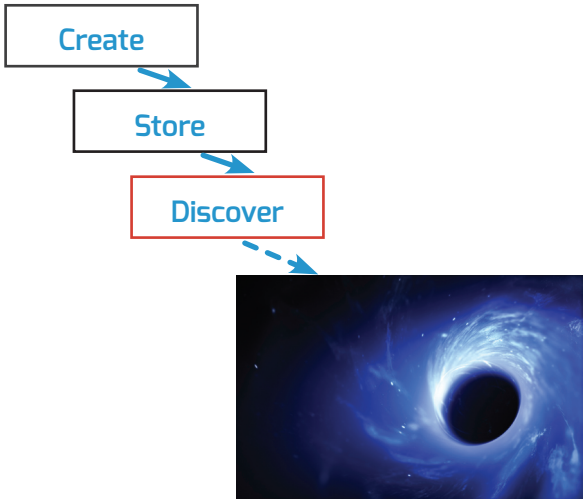
Why search is business-critical

Every day every single one of your employees will make decisions that could have an impact on the performance of the organisation and on their own careers. Some of these decisions may have a small immediate impact, such as a manager deciding on the agenda for a meeting. Others may have a very significant impact, such as whether or not to proceed with the acquisition of a competitor. Making a decision always involves ensuring that all the relevant information is available. The initial element of the decision process is often a period of learning to provide a context for the decision and ensuring that the best quality information is available is very important.

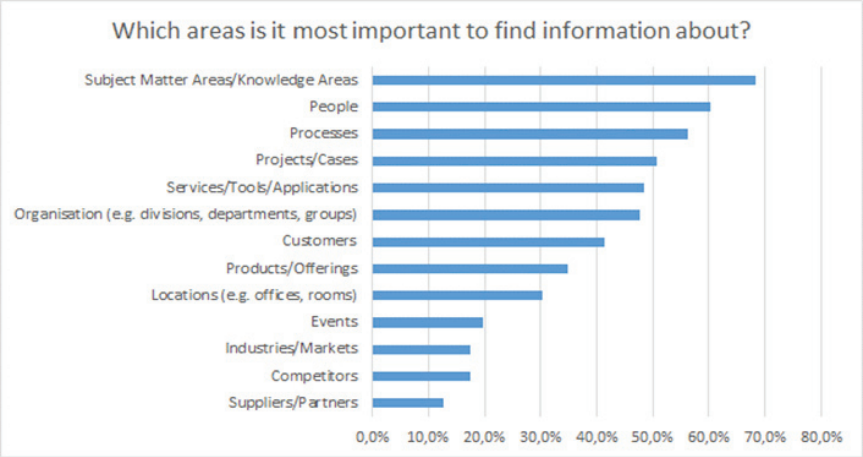
Information has a life cycle.



Creating information and storing it is quite straightforward. It can then be used by employees other than the author, who may themselves share it with colleagues. Ideally it should then be reviewed and either revised, added to a records management application or disposed of. In the centre of the figure above is the act of Discovery. If employees cannot find the information because the search applications are ineffective then in effect the information is invisible, ending up in the information equivalent of a black hole.



The chart below comes from research carried out by Findwise and provides a good indication of the range of information that employees are seeking. It is much more varied than Microsoft Office files stored in SharePoint. Some of it comes into the enterprise from external sources, such as market research and documents from clients, customers and suppliers.



Findwise Enterprise Search and Findability Survey 2016 www.findwise.com

Looking at this chart it may seem that information on topics such as industry and market developments is of comparatively little importance. This information may only be important to a relatively small group of employees but to them effective access to this information is business critical in terms of making business decisions about the commercial strategy of the organisation.

Use cases for enterprise search

There are three primary use cases for search:

Learning, in which the user may not be quite sure what the 'best' query is and will expect the search application to guide them through features such as auto-suggestion for queries. There is usually no time pressure for this learning process, and the user may return to the search application a number of times to accumulate all the information they need. This is often called exploratory search and the user expects the search application to provide most (and ideally all) of the relevant documents.

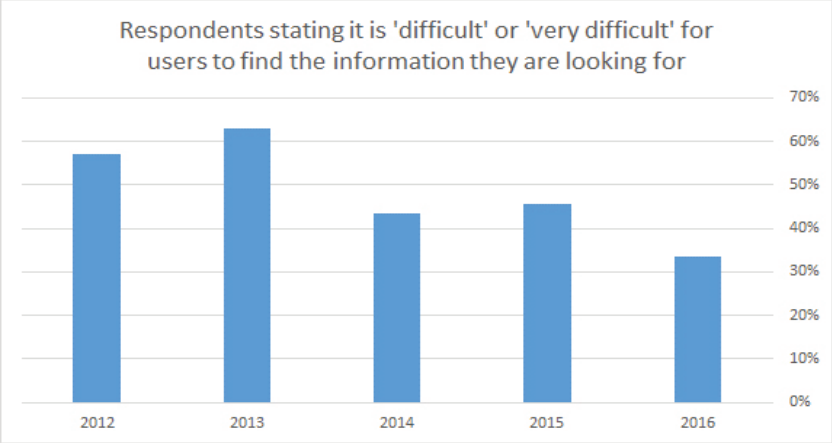
Task completion, where the user wants to find either a specific application to complete a task (initiate the recruitment of a new employee) or a document that provides detailed guidance on the process. The process may be different in different countries, but the user expects that either the application or the guidance document will be at the top of the results list even though they may have used a very short query such as 'new employee process' and not included their location in the query.

Reassurance, where the user is about to make a decision and wants to make sure that they have found not only the most relevant documents but also documents which might be less comprehensive but are much more recent. They may have the sales figures for Q1 and Q2 but need to make sure that Q3 data is included before making a decision.

A much more comprehensive categorisation can be found in Designing the Search Experience by Tony Russell-Rose and Tyler Tate.

The state of search

Despite the importance of being able to find information, very few organisations are able to satisfy the search requirements of their employees.



Findwise Enterprise Search and Findability Survey 2016 www.findwise.com

This is rarely because there is a fundamental problem with technology. The core elements of search technology date back over 40 years and current generation commercial and open source applications have well proven functionality.

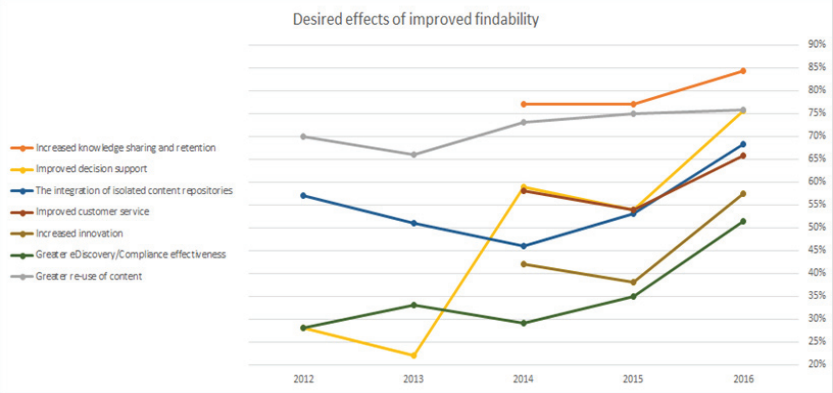
There are a number of reasons for this poor performance:

- It is impossible for a search user to know whether the reason that they cannot find a document is because the document does not exist, or the document is in an application which has not been indexed
- Search technology cannot overcome poor information quality, such as PowerPoint files being stored without clear titles
- Search requires a skilled support team to ensure that the technology is used to its maximum capabilities

Search applications are used by every employee in the organisation and need the same level of support as is given to other enterprise-wide applications. Reviewing and acting on search log information on how well the search application has delivered against specific queries is especially important. The scale of search use may not be immediately apparent, even to IT teams.

Making the business case for investment

Over the last five years Findwise (one of the largest search integration companies) has been tracking changes in the priorities and approaches that organisations are adopting in investing in improved findability. Since 2012 the importance of support for decision making has shown a very marked increase.



Findwise Enterprise Search and Findability Survey 2016 www.findwise.com

Before making a decision to acquire new technology it is always advisable to consider whether an investment in a search support team and perhaps some external consultancy might make a significant difference to the performance of the current application. This also applies to situations where a SharePoint implementation has been undertaken and the search performance is not as good as anticipated. The core functionality of SharePoint search is good, but as with all search applications it requires a team with specific search expertise to get the best out of it.

All IT investments need to be subject to a business case assessment. In most cases the business case is made on the importance of requirements such as meeting compliance requirements (financial systems), knowing the scale and location of inventory (ERP) or tracking employee information. An element of these business cases is an improvement in productivity.

Using productivity or efficiency as a business case is extremely difficult with search. Among the challenges are:

- There are never any reliable quantitative measures of productivity for information seeking
- It is difficult to determine the end-point of the task. Is it document clicked, document downloaded, document read, or document used?
- Different types of search (looking for a known document vs. looking for ideas) take very different periods of time
- Using a high-quality search application could mean that the time to complete a search is longer rather than shorter as the user has more options to refine the search and more relevant documents to review
- How can time saved on search be converted to a Total Cost of Ownership?

In the case of search there is no workflow element that can be taken into account in business case development. Some organisations try to make a business case on the time saved by being able to undertake searches more quickly but this is undermined by searches with good technology taking longer but providing substantially better results.

One effective approach to justifying investment in search applications and their support teams is to take a risk-based approach. Larger organisations will have risk managers who will report to the Board on any risks that could put the commercial success and reputation of the organisation at risk. This risk is scored, usually on the basis of probability and impact. The question for the Board is whether a lack of investment in search is putting the business at risk.

An information charter

Organisations often have policies that cover reducing environmental impact, employee relationships and corporate social responsibility. The following example is taken from Intranet Focus.

Our commitment to all our employees is that they can:

1. Find the internal and external information they need to make effective business decisions that reduce corporate risk, enhance the achievement of strategic and operational objectives and enable them to develop their careers.
2. Trust that the information they find is the best and most current available.
3. Publish information so that it can be used by other employees as quickly as is appropriate.
4. Locate and take advantage of the expertise and experience of other employees.
5. Link to internal and external social and business networks.
6. Be confident that the roles and responsibilities of their manager include ensuring that their information requirements are recognised and addressed appropriately.
7. Be assured that the organisation complies with all legal and regulatory requirements for the retention, use and transmission of information.
8. Take advantage of training in how to be effective users and managers of information resources.

The level of investment in search needs to match the importance to your organisation of employees being able to find the best available information with the minimum delay.

The technology of search

Martin White

Many of the elements of the technology that underpins search applications date back to the late 1950s. This technology is often referred to by the term 'information retrieval', a term still widely used by the academic research community. The major annual conference for the community is sponsored by ACM SIGIR (Association for Computing Machinery Special Interest Group for Information Retrieval) http://sigir.org which celebrated its 40th anniversary in 2017.

Although search is often cited as a technology, in reality it is blend of applied mathematics (especially probability and vector mathematics), computational linguistics and natural language processing and database management.

The diagram below presents a linear view of the process of search. Although highly simplified, it does set out the main elements of any of the search products on the market at present.

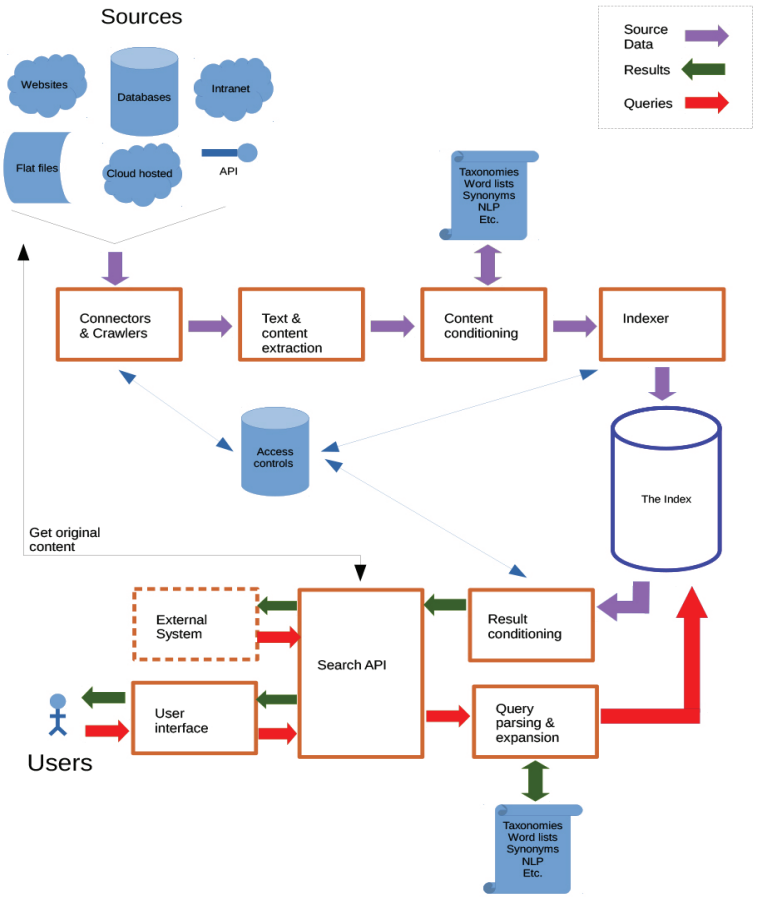
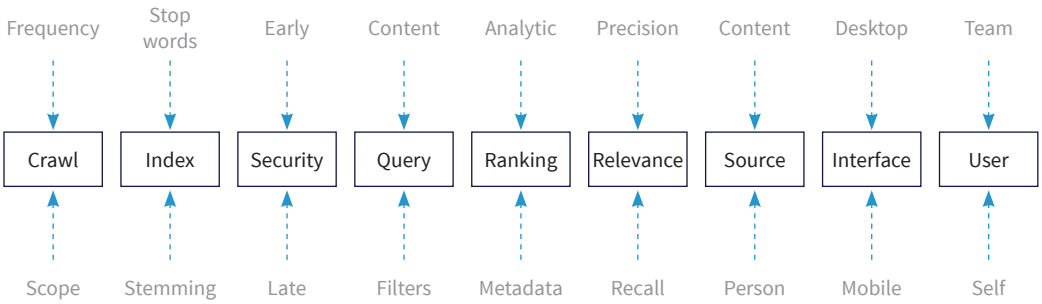


Figure originally published in "Searching the Enterprise", Foundations and Trends® in Information Retrieval: Vol. 11: No. 1. Reproduced with kind permission.

Search is an excellent example of the 'weakest link in the chain' model. Poor performance in any of these major elements cannot be made up through superior performance in others. Diagnosing the cause of poor performance (as measured by user satisfaction) can be very difficult to both undertake and remedy. This is why very high standards in defining user requirements, implementation management and user testing are so important.



The performance of any search application is highly dependent on the quality of the index, the latency of the delivery of search results and on the quality of relevance ranking.

The decisions that need to be taken around crawling include which repositories are going to be crawled, the requirements for interim crawls (incremental/push) and the frequency. Crawl frequencies in cloud implementations may be limited by the vendor and the extent to which these are optimal for the content and use cases of the organisation should be considered at the time of contract discussion.

In preparing the index the search application is undertaking a very significant amount of processing to convert linear text, tables and other content into a format that can then be matched against the query terms. Stemming and lemmatisation are both required to ensure that variants in spelling and language are indexed correctly. Requirements for multilingual and cross-language search need to be considered. As a wider topic it is important to be certain about when changes to the search functionality might require a complete re-index. The re-indexing may take a week or more to complete and check.

Security management is a very important feature. Employees should only be able to see information that they have permission to access. These access privileges are usually built into the identity management application, for example as an Active Directory record. This immediately raises the issue about who decides on these access rights.

There are two basic approaches. In what is termed 'early binding', the employee is only able to search through document collections, or documents, to which they have access permission. In the case of 'late binding', the decision on which documents an employee can see are made after the initial set of documents is prepared, with each document being matched against the current access permissions. In either case the processing involved can introduce a degree of latency into the search process, and this may vary between searches on open content and limited-access content. It is this variation in latency, for reasons for which the user is unaware, that can give rise to concerns about whether the search application is working properly.

Once the user has posted a query, the search application has to undertake a substantial amount of processing in order to do more than just respond with a list of results showing documents that contain the query terms. The query application has to be able to detect spelling errors, identify entities (e.g. that 56770-122-145 is a part code for a semiconductor chip that should have been written as SC56770/122/145), and suggest alternate query terms based on information gained from the index and from other searches carried out by the user and by others in the organisation.

The query may result in a substantial number of results, and most search applications will offer filters and facets to help the user refine their search. However, the choice of these filters and facets is critical. Offering a date filter requires clarity about whether the date is the date when the document was last modified or the date when it was first published. Both can be useful but not when presented in the same facet. This is also where consistent metadata and taxonomy management is very important (see Search and taxonomies, page 17).

Next comes the process of presenting a ranked list in order of relevance to the user. The fundamental problem is that two users with very similar levels of knowledge may have very different views on relevance. This issue is considered in more detail in The relevance of relevance, page 12.

As a result of any document having a wide range of characteristics, search user interfaces are often very complex. The view is often taken that search should be as intuitive as Google, but in reality it can take a substantial amount of trial and error to get the best from the options available on Google (including moving to Google Scholar, which presents a significantly different range of filter options). Employees need to be able to receive training in how to get the best out of any search application. If they are not able to find information on a specific topic then they need to feel certain that the information does not exist, and not be concerned that their lack of skill may result in an embarrassing meeting with colleagues where others have found information that they were unable to locate.

There are two implications to these complex user interfaces. The first is that people with visual and other disabilities find them difficult to use. Rarely can search interfaces be managed with the arrow keys, and voice-output readers are poorly supported. The second is that the interfaces do not easily transform to a mobile application, especially a smart phone. Search UIs are invariably difficult to degrade using responsive design code, and of course the user may be faced with reading a long document, usually with no keyword highlighting and with no ability to print out a document locally. The trend now is towards developing specific applications for smart phones, in particular people search, rather than trying to condense the desktop view onto the smart phone.

There is an outstanding set of blog posts by Daniel Tunkelang (formerly at Endeca and then LinkedIn) which provide short introductions to almost every element of search technology.

The relevance of relevance

Doug Turnbull

If search is answering users' questions, then relevance is answering them well. Even expert humans struggle to offer relevant answers. Getting a machine to do it for a given domain, in a particular business, with specific users in mind can be maddening. Amazon and Google have spent billions, and still have bad days.

Luckily, your search and relevance problems can be eased with far less investment than Google or Amazon.

Use cases - understand what you're optimising for

Search is used to accomplish a broad range of tasks. Fixing your relevance problems depends on what use cases you're solving for. Take a catalogue of how your users use search. Examples of use cases include:

- **Navigational search:** users looking up an item by its name. The search in the contacts on your phone, a form by a form number
- **Informational search:** hunting for a fact. Searching for tomorrow's weather, or your 'frequent flyer miles' number in your email
- **Compare / Contrast:** deciding between a contrasting set of options. Comparing flights, products, jobs that fall within the search criteria
- **Category search:** an overview of a category. Searching for "sneakers" highlighting the sneakers sold by the store. Or "part time" showing the best part time jobs.
- **Exhaustive research:** collecting many bits of information about a topic. Searching and reading articles on "child language acquisition in French Guinea", or researching patents similar to your idea.

These are just a sample. Each use case has unique ranking solutions. Most search applications have several use cases, competing for prioritisation.

Tune beyond field weightings

As Charlie Hull points out, "boosts are considered harmful". Optimising relevance, with conflicting use cases, cannot be achieved by setting weights on fields.

Consider a movie search example. When users search for an actor, one might prioritise the recent movies starring that actor. However, when searching for a movie title like "Star Wars," a user might actually want the opposite: i.e. the very first Star Wars film. In other words, the weighing of ranking factors completely depends on the search use case.

One strategy to achieve this might be to (literally) target relevance scoring for each use case: "IF (strong match on field) THEN (apply scoring formula)". Perhaps for actor names. A "strong match" might be a phrase match on the cast field. For example, matching the full phrase "sylvester stallone." Something approximating this strategy in Solr might look like this:

```
q=sylvester stallone
strongActorMatch={!edismax qf='' pf='cast' v=$q}
actorSort=recip(ms(NOW,release_date),3.16e-11,1,1)
bf=if(query($strongActorMatch),$actorSort,0)
bf=if(query($strongTitleMatch),$titleSort,0)
```

Solr syntax can be cryptic. Don't worry about understanding the exact syntax. The upshot is that "strongActorMatch" will result in a relevance score if a phrase match occurs on the query. This satisfies the if statement in "bf" (a 'boost function'), and triggers the "actorSort" formula to score results. Here actorSort is a classic Solr date boost formula from the project documentation. The rest of the "Title search" example is omitted, but shown to demonstrate layering in an additional use-case dependent boost. More detail on boosting strategies targeting for Solr/Elasticsearch can be found in the book Relevant Search (http://manning.com/books/relevant-search).

Test-driven relevance

Relevance testing differs from other forms of automated testing. Search does not simply pass/fail. Search always exists in a grey area, hopefully trending upwards in quality. Measuring search quality in tests requires gathering a judgement list. A judgement list corresponds to an assigned grade for a document for a query. The table below shows an example judgement list for movies:

Query	Movie title	Grade
Star Wars	Star Wars: A New Hope	A
Star Wars	Star Wars: The Empire Strikes Back	B
Star Wars	Star Wars: Return of The Jedi	B
Star Wars	Star Wars: The Last Jedi	B
Star Wars	Spaceballs	C
Star Wars	Star Trek	D
Star Wars	Sense and Sensibility	F
Star Wars	Rambo	F
--	--	--
Star Trek	Star Trek	A
--	--	--

With this set of reasonable grades, if we search for "Star Wars" which set of search results below is better?

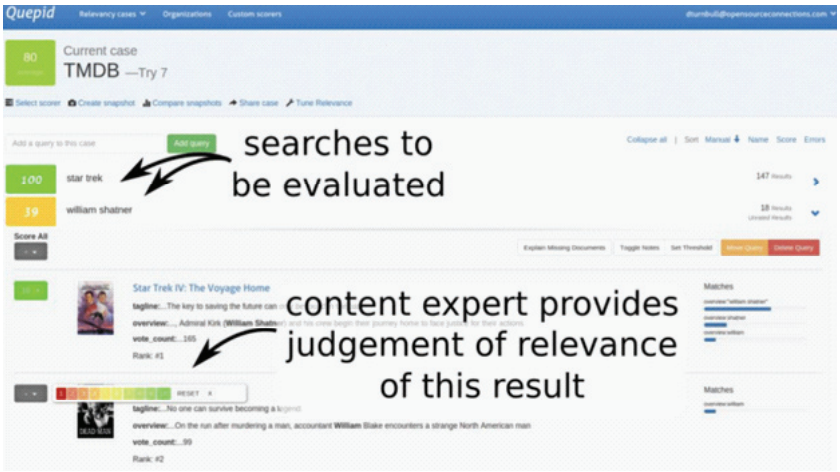
Results 1	Results 2
1. Spaceballs(C) 2. Star Wars: A New Hope(A) 3. Star Trek(D) 4. Sense and Sensibility(F) 5. Star Wars: The Last Jedi(B)	1. Star Wars: A New Hope(A) 2. Spaceballs(C) 3. Star Wars: The Last Jedi(B) 4. Rambo(F) 5. Sense and Sensibility(F)

On the one hand, set 2 has more relevant content crowded towards the top. However, set 2 also has two duds in the top five. Assigning a good quality score can be hard to do – and is a constant source of healthy debate. Indeed, it depends on the use case, the application, and the organisation.

A series of classic search metrics evaluates a set of search results given a judgement list. Normalised Discounted Cumulative Gain (NDCG), while a mouthful, is very popular. It measures, on a scale of 0-1, how far the current result set is from an ideal. Further, it incorporates a position bias: the maths punishes result 1 being out of kilter far worse than result number 20!

Gathering judgements can seem exhausting. Some teams build judgements from analytics data. Some rely on experts to manually grade search results as good/bad. Others incorporate crowdsourcing to grade results. Grading results need not be exhaustive. Covering sufficient examples from the most important use cases can deliver tremendous value.

Finally, tooling can help. Products such as Quepid (shown below - disclaimer the author's company develops Quepid) and open source tools such as Netflix's search test framework (https://github.com/Netflix/q) can ease the work involved.



Gathering relevance judgements on Quepid

Get thee to a taxonomy

A common relevance problem comes from managing synonyms. Search comes with the vocabulary problem -- where two human beings speaking the same language use different phrases to describe the same item .

Most search engines have an ability to deal with this problem through a synonym functionality. Making "trousers" and "jeans" equivalent causes "trousers" searches to return blue jeans. Solving one problem though, creates another. With this equivalency, searching for jeans now has the effect of returning other kinds of trousers (perhaps khakis) higher than the blue jeans!

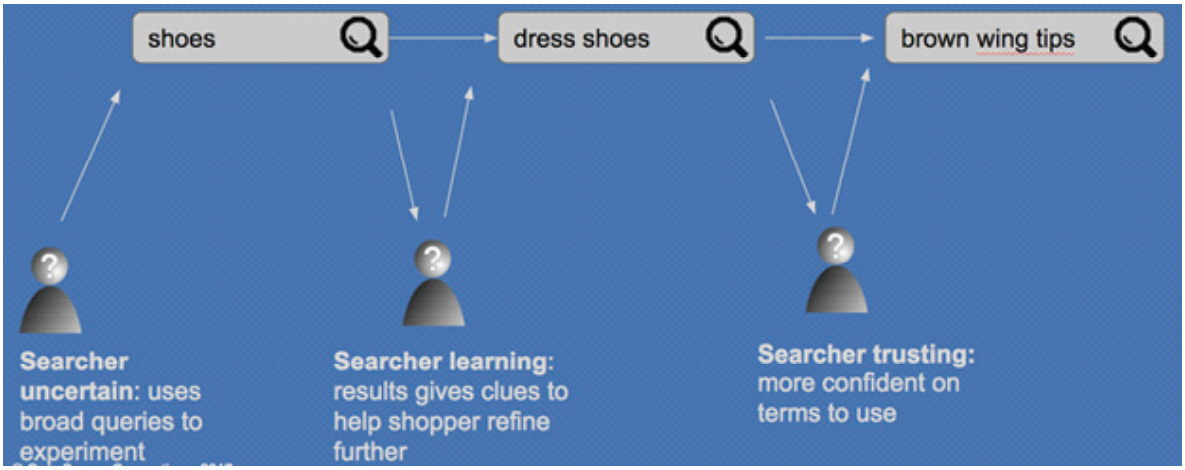
Jeans are a kind of trousers. Instead of a synonym, they are a hyponym of trousers. Search teams often mistreat hypernym/hyponyms as synonyms, creating unexpected equivalences (like our jeans example). Search teams who depend heavily on synonyms likely really need a taxonomy. A taxonomy is a hierarchy of concepts, each identified

with a label -- the phrase that describes the concept. Concepts have alternate labels, very strict synonyms. For concept “Graduation Ceremony”, a “Grad Ceremony” could be an alternate label. Each concept has a hypernym, a broader parent concept, and hyponyms, more specific concepts.

Taxonomies give search teams tremendous power. For example, a very common strategy is to expand the query at query time to direct hyponyms. A query for “trousers” expands to “trousers OR jeans OR khakis OR capris …” producing the desired effect of bringing back more types of trousers. A search for the child concept “jeans” just searches for “jeans.”

Taxonomies give the opportunity to guide the user when we don’t have what they want. If we know “oxfords” are a kind of dress shoe, but no oxfords exist in the catalogue, then optionally we can fall back to a search for the hypernym dress shoes. Or we can prompt the user to make this expansion themselves (for more information on taxonomies, go to the chapter by Helen Lippell, page 17).

Query logs can be a ripe source for building a taxonomy. Users often strike out with broad queries, then refine down to more specific hyponyms, as shown in the diagram below . Algorithms like SHReC can assist (though never replace) an information scientist in discovering concept relationships from search logs and content.



Those machine learning silver bullets

“Cognitive search” is discussed a great deal in the search world these days. Should you take the plunge? Is the complexity worthwhile? Is it the silver bullet to relevance?

Here are some questions to ask yourself before continuing:

- Do you have organisational sophistication in data science to understand, manipulate (and sometimes get under the hood) to improve machine learning algorithms for your domain?
- Do you have experience of gathering and developing training data from users? Is the data good enough to act as a training set?
- Is the value from search worth expanding your search team significantly, or spending on a vendor?

Machine learning takes more sophistication than hand-tuned relevance. But it can offer more value. It's best to start with manual methods. Hire a good taxonomist and relevance engineer. Then decide if machine learning is worth the investment. The taxonomy, boosts and other strategies aren't discarded when moving to machine learning. Instead they form a foundation to get to the next level of relevance.

Search and taxonomies

Helen Lippell

Search engines can't read your mind

Taxonomies and metadata play a critical role in search relevance and user experience. User aspirations for their enterprise search systems have been steadily rising over the past few years. When told that there will be a project to improve their company search function, users often say, "Just make it like Google!" They want to put a few keywords into the box and for search to totally 'get' what they're looking for as if by magic.

Google has the benefit of huge amounts of personalised data about every user, engineering capacity at immense scale, the corporate power to boost or remove content as it pleases, and the ability to manage thousands of different parameters about content quality. Most organisations simply don't have this firepower to call on. But if you're a hard-pressed search manager, it's not all bad news. Taxonomies and metadata are an important component of search success, and you don't have to be a tech industry behemoth to make them work for you. A search engine application is a 'blunt instrument' but taxonomies add refinement and detail.

From basic tagging to taxonomies and ontologies: why do any of it?

In essence, taxonomies and metadata codify your and the business's understanding of user needs, user language and content structure, into controlled vocabularies and schemas. By using them in your search, you can improve people's experience and help them do their work more comfortably.

To get around a poor search system, users develop all sorts of hacks to help them capture and organise information. For example, they might use draft emails to store bookmarks, or create folders called "My Stuff". These techniques often arise because people lack confidence in enterprise applications.

Improving the quality of search across the information resources they need the most should reduce the need for workarounds. Even a basic tagging scheme could add contextual metadata such as date validity of the document, intended audience or basic synonyms which connect corporate jargon to users' mental models of what they are looking for. More complex taxonomies, content models and ontologies take it further by enabling more sophisticated relationships between concepts to be derived without the content creator or consumer needing to do any extra work.

How metadata can enhance the search user interface

Most enterprise search systems take user input from the familiar single text box (as opposed to more complex interfaces which make users select filters and parameters before sending the query to the search engine). This means that the results interface should make full use of metadata and classification to offer users straightforward ways to improve their experience. Facets and filters are very common on e-commerce search pages, and in a corporate setting, facets and filters enable people to build up sophisticated queries. These might include information such as content format, date, site section, tags and so on.

Unlike general web browsing for entertainment or leisure, corporate users will have specific needs. It is much harder for them to 'satisfice', that is, to make a choice that isn't the best available but is 'good enough' for its purpose. If they need a particular

policy or strategy document, they can't just look at something similar from another source, as they can do on the web when researching their holidays. This is why the extra information added through metadata and taxonomy classification helps save time and frustration.

Structured content (e.g. web or intranet pages)

If you are starting from a position where your content management system is capturing little or no metadata, it is helpful to find any structure that exists in the content itself, (e.g. mark-up for title, description, created date etc.) and use that as a signal within the search. A good search engine will have back-end tools or APIs to use structure in the index, so that important parts of pages can be boosted to improve relevance. Content authors should be trained to optimise their titles and description fields for search once this has happened (this might seem obvious, yet although the Web has been around for over 20 years, it is still needed!).

It is always easier for search engines to be effective against structured pages rather than documents produced in formats such as PDF, Word or Excel. Even Google returns weaker results for public content which is locked away in PDFs. Documents do not have the same degree of metadata options that a content management system does, and it is easier to apply controlled taxonomy tags to a CMS page than to an uploaded Word document.

Unstructured content (e.g. enterprise social networks)

Enterprise social networks (ESNs) have gained traction over the last few years and are now an important part of the digital workplace. This presents a challenge for traditional search and taxonomy technologies, as the content is unstructured and less controlled than say, intranet content. Synonym management becomes even more important than on a corporate intranet, as ESN users won't necessarily be trained content authors who know the 'correct' terminology for a policy, product etc.

Automatic tagging against a human-created taxonomy can help bring order and relevance for anyone searching across ESNs and messageboards. Automatic tagging enables classification at scale, but it is never cost-free in terms of the need for human oversight. People spot things and nuances in context, which computers are still relatively weak at. (Even the leading AI platforms rely on a huge amount of human curation, and these are still beyond the reach of most normal organisations and hard-pressed information professionals.)

Supporting the business in improving metadata and taxonomies

The most important success factor for an enterprise search project is the quality of engagement with the business (ahead even of the technology used and the content created). Clearly, for any project to be successful, technology, content and business factors must all be delivered to a high degree of quality. However, without efforts to support the people and processes that depend on search, the positive impact of improving the technology and the content will be reduced.

If staff will be applying or reviewing metadata, then this part of the content authoring process should be made a key part of their roles, not the 'bit you have to do once you've done the writing'. This is important no matter who the users are. Whether your users are journalists, video game developers or police officers, they will probably need persuasion that this metadata stuff is important and will be contributing to a wider improvement in how the business manages its information.

During (and indeed, after) the project, taxonomy developers/managers should be visible and communicate regularly within and outside the project. In an organisation where taxonomy is new, education and reassurance are needed. Conversely, where taxonomy is well understood, managing a narrative around expectations might be needed to avoid disappointment that taxonomy cannot fix every single issue with search, and isn't omniscient AI either!

Avoiding digital entropy by keeping taxonomy and content quality high over time

Governance of a taxonomy should not be merely a couple of sides of A4 added at the end of a development project, so the project manager can tick a box on a checklist. The best governance processes always find a 'Goldilocks' position between involving the right people from across the whole user base and keeping the number of meetings or change control forms for these stakeholders down to a manageable level.

Ongoing maintenance is critical, because every organisation changes - new products are developed, departments restructure, business language changes and users change. A taxonomist needs to be empowered with time and resources to update the taxonomy regularly. It does not matter how sustainable the original taxonomy was intended to be. Well-constructed taxonomies can fall into disuse and be in need of a major overhaul - when it would have been less work to keep the taxonomy relevant on an incremental basis.

Finally, content might not be directly under the purview of the taxonomist, but taxonomy managers should work closely with colleagues in content production and strategy. The main benefit of this is to be aware in advance of new content, campaigns or initiatives which will have new jargon/keywords to add to the taxonomy. Taxonomy managers can also give guidance on content structure and tagging.

A note on taxonomy management tools

Many organisations, even those with robust information management processes in other areas, are still managing taxonomies in a spreadsheet. Even worse, they might be managing multiple unconnected taxonomies in multiple spreadsheets. This can and does lead to errors, lack of business adoption and loss of data if the ad hoc taxonomist leaves. I encourage organisations to look at using a specialist taxonomy management tool instead.

The largest hurdle is usually acquiring budget, especially if there is a content management system (CMS) or SharePoint implementation that has some in-built metadata management capability. However, the best taxonomy tools on the market have a number of key advantages over these:

- They support the full range of taxonomic relationships (broader term, narrower term, related term, scope note, used for/used, also polyhierarchy), not just a simple parent/child tree structure
- The more advanced tools support ontology management, and linked data formats
- Unlike a spreadsheet, it is much harder to introduce data integrity issues such as circular relationships or duplicate labels
- Full administration features such as versioning, change logging, access control and user account management

A potential issue compared to using a CMS or a SharePoint (or equivalent) system is setting up the right data integration between the taxonomy tool and the consuming application, whether this is a publishing system or enterprise search. However, modern tools have a wide range of capabilities with regard to exporting and integrating data, so this should not be a barrier to successful adoption.

Vendors

The following vendors are well-established companies that sell taxonomy management tools (plus VocBench, which is EU-funded, and free). They all offer high-quality products and services. The 'best' solution for any project, of course, will depend on exactly what functionality is needed by the taxonomist, and what budget is available.

Data Harmony products by Access Innovations
http://www.accessinn.com/products/
Multites
http://www.multites.com/
PoolParty
https://www.poolparty.biz/
Synaptica
http://www.synaptica.com/taxonomy-management/
TopBraid products by Topquadrant
https://www.topquadrant.com/products/
VocBench
http://vocbench.uniroma2.it/

Working with commercial vendors

Miles Kehoe

This chapter covers commercial vendors of proprietary product companies as well as companies whose product is based on open source, but which enhance the capabilities with their own proprietary extensions.

Acquiring any enterprise application is difficult. Enterprise search, perhaps more than any other enterprise application, potentially touches every document in the organisation, from file shares, databases, content management systems and custom applications. If that was not complex enough, it is highly unlikely that the IT department will have any previous experience of specifying, selecting and implementing a search application.

Start with a strategy

The process of specifying, selecting, negotiating, evaluating, installing and then implementing a new search application is likely to take at least a year (see Planning the project, page 37). Success depends on keeping a balance between the tactical requirements of the project and the vision for the impact the application will have over the next three years. Without a strategy there is unlikely to be clarity about the outcomes and as a result the project will lose focus and probably come to a premature conclusion.

A search application not only delivers to potentially every employee but interfaces with a range of other applications and has to conform to corporate IT standards. Appendix B sets out the scope of a search strategy in an A-Z list of topics, to avoid inadvertently prioritising some over others. There are 50 topics in the list. Not all might be relevant but even the process of discarding some may provoke a lively discussion.

In an ideal world this strategy should be owned by a Search Steering Committee that is broadly representative of the business.

Creating a search evaluation team

Because search touches so many parts of the organisation it is essential to enlist a team of individuals from across the organisation who will assist in defining requirements and be involved (even if indirectly), in the selection and evaluation process. They will also communicate the team's activities back to their functional areas to help ensure that their requirements and expectations are considered. There may be other search applications in the organisation, and these should also be represented on the project team.

Bringing in experience

If you have a strong internal team familiar with the enterprise search market, you may be successful 'going it alone'. If not, you may consider engaging an external consulting firm that specialises in enterprise search implementation. Some firms work with a single search vendor but at this point you are casting a wide net. Finding a firm or individual that works with a number of commercial (and even open source) vendors may be better for you at this point in your journey. The firm you engage should be able to work with you to gather and organise your requirements; recommend vendors for review; facilitate an introduction to those vendors; and assist in the selection and implementation of search if required.

Gather requirements

Some of the obvious requirements for enterprise search are relatively easy to identify and document, such as identifying current repositories, security, remote access, and the standard hardware your organisation uses. What is more difficult is gaining a balanced view of user requirements, linked to business requirements. Search projects are often started because one senior manager had a poor search experience and wants to change to Google! One person, no matter how senior, is not representative of the entire employee base.

It is important to appreciate that the greater the extent of personalisation of the search results, the more certain you need to be that the personalisation profiles can be defined in enough detail for the algorithms to be written, and the more time you will need to allow for testing and accepting the algorithms during the implementation phase.

Your search evaluation team can find out from their colleagues what works, and what does not, with the current search application. They may also have suggestions for additional capabilities that have widespread appeal. Remember that great ideas can come from unexpected sources. Be sure to ask the people you and your team interview what other things they would like to see search provide. It can also be valuable to talk to people who have recently joined the organisation as they may have come from an organisation with a well-implemented application. They may also be searching for information that established employees know how to find.

Vendor selection

You need to be working towards having a single vendor in the validation (proof of concept) stage. Running either parallel or consecutive validation processes is guaranteed to throw the project way off schedule. The objective at this stage is to cast a wide net and then to narrow the field down to two or three (at the most) vendors for deeper evaluation.

Your initial contact with a commercial search firm will likely be a sales representative. At this point, describe your requirements in general terms but do provide sufficient information so the candidate vendors can focus on how their product will solve your specific search needs. A demonstration is always interesting but unlikely to be of value because search is a platform, not a product, and seeing how well it works on test collections curated by the vendor is no judge of how well it will meet your specific requirements.

Bear in mind that it may be a year or more before the integration is completed, so an indication of their technical roadmap is essential. The timing is important because the last thing you want to be doing is selecting a vendor on the basis of the current version, running a proof of concept on the next version of the software and then having to implement a major upgrade. Ask for two references you can talk to and use your social media communities to find other users of the product to speak with.

RFI and RFP

The standard procurement route is to develop an initial Request for Information (RFI). Based on an analysis of the responses to the RFIs a Request for a Proposal (RFP) is prepared. The list of features of an enterprise search product may well be several hundred items long. The danger is that the application meets the specification, but it does not meet business requirements. Only when the application meets the content can the value of the functionality be assessed. A better solution would be to set out core/quasi-mandatory requirements in an RFI and then eventually work with the selected vendor on a mutually-developed RFP which is in effect an outline of the contract.

Integration partners

Many vendors do not provide the full suite of professional services you are likely to require. They will work through selected partners, especially if the implementation is a global project and the vendor is only able to support implementation in their own country or region. Because of the complexity of the implementation process you should be selecting the vendor + implementor as a package. This can sometimes cause confusion when it comes to contract negotiation so be certain about respective responsibilities as well as the lines of communication between the vendor, the implementor and your project team.

Features vs budget

The outcome of the RFI stage should ideally be three vendors:

- The preferred vendor (Vendor A)
- The next best vendor (Vendor B)
- A third vendor just in case either Vendor A or Vendor B fall by the wayside. You should always have an option.

Now is the time to request a price estimate from each vendor, including software licensing, support, training, and implantation assistance, unless your staff or another firm can do that work. Be sure to review each vendor's license terms (some licenses may be a three-year term and others may be perpetual). The total cost of ownership can vary widely. Remember that price and terms are almost always negotiable; if your preferred vendor doesn't show flexibility, it may be worthwhile talking with the company whose product finished second. Realistically the best outcome you will get is a Rough Order of Magnitude, but that should be enough to make sure that time is not wasted when the cost is likely to be way out of line with the budget. However whoever in IT set the budget is unlikely to appreciate what good search applications may cost to implement, which is why we have included a chapter on budgeting a search project, page 47.

Onsite evaluation

In theory it would be useful to be able to undertake a validation exercise with perhaps two or three vendors for deeper evaluation. A decision has to be made whether the vendor/product evaluations take place concurrently or in sequence. Concurrent evaluations will be very resource intensive, and it will be important to silo the teams. That means separate offices and splitting the project team into two or three groups, one for each vendor. Sequential evaluations could mean that the entire evaluation process might take six months. By that time the first vendor may have launched a revised version that more closely meets your requirements. You also need to factor in the costs that the vendor may charge for a validation exercise, especially if assessing personalisation is going to be important.

If your organisation has concerns about validating just one vendor an option could be a schedule that ensures the most important features are assessed first and it is made clear to the vendor that if they fail these then the rest of the evaluation will be terminated.

Select a number of repositories that are representative of the type of content you have and the security approaches you utilise across the organisation. If your organisation allows content to reside on cloud systems, it may make sense to evaluate both on-prem and cloud solutions.

Some other things to consider:

- Installing the software and the initial index should be a joint operation between your IT team and the vendor
- In evaluating result quality, consider both precision (finding a specific document) and recall (finding all documents on a specific topic)
- Ensure employees who actually use the various repositories you've indexed are helping evaluate the quality of the results
- Ask appropriate IT and search users to utilise the reporting components of the product; understand if and how the reports can be expanded or customised to meet your reporting requirements
- Invite members of the search team as well as other employees at all levels of management to actually use the demo to get a feeling for the capabilities and interface
- Confirm that document-level security is working correctly across multiple repositories

This process can take some time depending on the complexity of the content and enterprise architecture and the schedule is very difficult to forecast.

Training and support

An important part of your vendor evaluation should include the process and effort of the eventual implementation and roll-out, including training and support costs. Ensure each vendor includes the costs of training and support in their proposal.

Ongoing search management

In the same way that you enlisted a team to assist in the vendor evaluation, the best run enterprise search implementations will often create a formal or informal advisory team to serve as liaison between the user community and the group who eventually manage the search project in IT or Operations. Members of this group (often known as a Search Centre of Excellence or SCOE), serve as ambassadors to their respective departments, and provide valuable feedback to the group responsible for ongoing search operations.

Usually individuals who participated in the original search selection and evaluation are willing and able to assist in the use of the new search and gather feedback from their departmental peers and even provide some assistance to their peers in the use of the new technology. Their feedback to the group managing the search implementation can provide valuable insight on the enhancements users want and the problems they are experiencing.

Working with open source search

Charlie Hull

When considering the available options for implementing a search application, open source software can seem like manna from heaven: freely available to download, open to worldwide scrutiny for potential security issues, packed with cutting-edge features, developed by hundreds or even thousands of programmers, with no vendor lock-in. The question is often "Why not?" rather than simply "Why?". However, adopting an open source route comes with its own set of unique challenges and risks.

The first thing to realise is that 'freely available' does not mean 'cost-free'. Like any search engine software, open source search will require server hosting, integration, customisation, tuning, support and training. However, the open source model gives you significant freedoms and control of your own solution, future-proofing you against the whims of vendors or changing fashions in software.

Don't go it alone unless you can afford a serious investment in your own in-house team, both in terms of wages and time. This is complex software that requires specialist skills that cannot be gained overnight. Open source search is usually less well-documented than closed source solutions and the documentation does not always keep up with the rapid pace of development. Developers prefer writing code to documentation and open source developers are no different.

Luckily there are many books, blog articles, how-to guides and presentations available on open source search. However, it can be difficult to be sure which of these resources is up-to-date – an online guide for version 5 of a certain software library may not work for version 6 due to a changed API or deprecated feature, and the documentation for this change may be incomplete or simply missing. Another good source of information is conferences and more informal gatherings such as Search Meetups, where your team can meet others with experience of the same challenges you are facing.

There are several routes one can take to make the process of adopting open source search less painful. One is to buy a commercial product based on the open source library you have chosen, where the vendor will have chosen the right version of the library and built its own closed source additions – for example, administration tools, connectors or user interfaces. Although this approach leverages some of the advantages of open source it does tie you into the vendor's roadmap and business model to some degree, removing some of the freedom of open source adoption.

Another option is to engage a specialist consultancy (often staffed by committers to the open source project - those who build and maintain it) who will have deep experience of the software and know exactly how best to implement it. Working with a trusted partner like this will give you all the advantages of the open source model and should also help your own team rapidly gain relevant skills and experience.

If you decide to end the partnership at some point you should then be able to go it alone (assuming you have allowed for suitable training and mentoring of your own team and made sure documentation has been provided and kept up-to-date). Most of these partners will deliver any modifications and additions to the core library as open source code (this doesn't mean you have to open up these additions to the world and possibly your competitors – but always check the software license that applies!). Note that most specialist open source consultancies are small companies who are highly adaptable and agile by nature.

Entering the world of open source also brings with it a set of commitments. The open source software you depend on itself depends on the efforts of many people who believe that open source development is the right thing to do. You can simply take the software as provided and build your solution without ever giving anything back to the community (and many do). However, it is preferable to engage with the community. Share any improvements you make, improve documentation, write blogs and articles, present and/or sponsor events and generally behave as a good open source citizen. After all, you gained so much from the efforts of others! This has the pleasant side-effect of raising your profile amongst the developer community – attractive to potential employees and increasingly to investors. If you encourage your developers to participate in this way, they will have some influence as to the future direction of the project which may further benefit your business.

Be aware that many closed source, commercial companies see open source as a clear threat to their business models – and it is. Competing with something that appears to be free is difficult. They will thus attempt to raise the spectres of Fear, Uncertainty and Doubt (FUD) – how can you trust this software when you don't know who's in charge of it? Who do you sue if it goes wrong? Surely being open makes it less secure as anyone could hack it? You may also have to defend the open source model against detractors from your own organisation. Suffice it to say that many governments now actively require consideration of open source software for publicly funded projects and many of the world's biggest companies have built their entire infrastructure on open source code. In recent years there have been many cases of commercial search engines being acquired by larger players as they realised their overpriced solutions are no longer competitive in a world where open source is in the ascendant, and many leading commercial vendors now build their solutions on an open source core, realising that reinventing the wheel is somewhat pointless. There are examples of open source search applications with indexes of billions of documents and millions of users.

To conclude, open source search is not without its risks and challenges. Open source is all about community – and the path to success depends on identifying the right people to build it, both within and outside your organisation.

The landscape of open source search

There are many search engines that have been released as open source search: however few of these have been widely adopted. Some are popular with the academic community as teaching and research tools but almost unknown to industry e.g. Lemur, Indri and Terrier. Nutch, although sometimes regarded as a search engine, is more focused on web crawling. Xapian is a powerful library with probabilistic ranking but has not been adopted widely (partly because its GPL license precludes embedding within commercial products).

The Apache Lucene library, created by Doug Cutting (who also created Nutch and Hadoop) in 1999, is an information retrieval system written in Java. Although it is possible to develop a search application directly on top of Lucene, this requires a deep knowledge of information retrieval and in recent years most open source search applications have used either the Apache Solr or Elasticsearch enterprise search servers, both of which are built on top of Lucene. The Lucene and Solr projects were merged by the Apache Foundation (a non-profit organisation which manages a huge range of open source projects, all of which use the permissive Apache License) in 2010 and the combined project is properly known as Apache Lucene/Solr. Elasticsearch, written by Shay Banon in 2010 and based on his earlier Compass search engine, has grown in popularity especially for log analysis applications.

Although many people have written detailed comparisons of Solr and Elasticsearch (and each engine has its fans and detractors) both are very similar in terms of capability and features. Each is accessed via HTTP APIs allowing developers to write interface code in a variety of programming languages. Both engines are highly scalable (indexing potentially billions of documents) using distributed computing and storage, to provide high availability, high performance and automatic failover capabilities.

Some technical differences exist: Solr uses the Apache Zookeeper project to handle its distributed model, which is a well-regarded method used by many other projects. Elasticsearch uses its own clustering algorithms which have in the past been shown to contain potential flaws (such as the "split brain problem"). However, when designed and implemented correctly, stable and performant applications can be built with either. In recent years both engines have acquired analysis capabilities allowing for complex calculations to be performed on the indexed data and associated projects such as Kibana can be used to graph and visualise the results. Elasticsearch in particular is popular for indexing and analysing huge amounts of log data.

It is also useful to compare the communities around each engine. Solr is managed by the Apache Foundation and any changes to the released code are carried out by 'committers' who are invited to join the project by the Lucene Project Management Committee (PMC). Bugs and issues are managed via a public instance of the JIRA system and anyone can submit code patches or improvements – but only committers can push these changes into the next version. We can thus consider Lucene/Solr as a truly open source project with open development. Committers and PMC members (who are usually also committers) work for a variety of organisations including huge IT companies and single-person consultancies – no single commercial organisation controls the roadmap and development. The largest employer of committers is Lucidworks (previously Lucid Imagination), founded by a number of Lucene/Solr committers, who also produce a commercial product Lucidworks Fusion which is based on Apache Lucene/Solr and Apache Spark. Lucidworks also provides support contracts and training for Lucene/Solr and runs an annual conference, Lucene/Solr Revolution.

Elasticsearch and a number of associated projects (ingestion tools Logstash, Beats and visualisation platform Kibana) are available as open source, but development is controlled by a commercial company founded by Shay Banon, Elastic. Only Elastic employees may commit changes to the code. Elastic sells support subscriptions which include licenses for a number of closed source products (the 'X Pack') which extend Elasticsearch with administration and security tools (it should be noted that open source alternatives exist for many of these). Elastic also provides training and runs its own annual conference, Elasticon as well as a number of smaller events.

In March 2018 Elastic, the company behind Elasticsearch, announced at their annual conference that they were 'opening up the code' for the X-Pack. It should be noted that this 'open' code does not meet the official definition of 'open source' (which Elastic has admitted) as the license to be used will be written by Elastic themselves - at the time of writing the license has not been made available. Users and contributors to the code may still have to pay to use some or all of the X-Pack features, and the exact level of 'openness' is still unclear and may be confusing to end users.

In addition to these large players, there is a varied ecosystem of smaller organisations (often run by or employing committers) providing consultancy, training, support and additional software compatible with Lucene/Solr and Elasticsearch. Many other software projects both open source and commercial embed these engines to provide a

search capability. Depending on budgets and other considerations end users may engage in commercial relationships with Lucidworks, Elastic or any of the smaller players or may choose simply to download the code and go it alone. Mailing lists, IRC channels and web forums provide community support on a non-SLA basis and there are many books, blogs and articles providing further information. Many small events such as Meetups occur around the world where developers and others can present their projects and ideas and network with others facing the same challenges.

The pace of development of both Lucene/Solr and Elasticsearch has accelerated over the last few years, partly fueled by venture capital investment in Lucidworks and Elastic. It cannot be denied that Lucene-based search engines are hugely popular and used by many thousands of organisations across the world. It is unlikely that any closed source commercial search engine can claim adoption at anything like the same scale.

An interesting recent addition is Vespa, released as open source by Yahoo in 2018. Vespa powers many of Yahoo's search features and can also be used to build recommendation systems at large scale. Although it contains many innovative features in particular for search ranking, it remains to be seen whether Vespa will gain community support and thus as wide adoption as Lucene, Solr and Elasticsearch.

Enterprise search as a service

Miles Kehoe

In the last few years, 'the cloud' has become a hot topic, and is billed as the ultimate solution for nearly every imaginable computing need. Enterprise search is no exception, and lately there has been a push on the part of most enterprise search vendors to offer a 'cloud-based solution'. Ironically, at least two companies – Atomz and Search-Button.com - introduced the concept as far back as the 1990s.

The most common approach today goes by the name 'cloud-based search'. The other approach, which has actually been around longer, is what is known as 'hosted search'. The two approaches are similar, but there are some subtle – and key - differences. It is important to appreciate that cloud and hybrid versions of on-premise software may have different functionalities in both the index and query management stages. This is because the cloud and hybrid versions have been optimised to take advantage of cloud technology, a much better approach than just taking on-premise software and loading it onto cloud servers.

Cloud-based search

In this model, standard commercial search software is installed on remote servers like those provided by Amazon Web Services (AWS). Whether the servers are maintained by the hosting company or by an internal IT team, access to the software is via a secure connection. Your corporate search team is responsible for understanding the software, including all aspects of managing a complex enterprise software product. The company providing the servers is responsible for installing and monitoring the systems.

It is important to note that some cloud vendors have taken great liberties with the term 'cloud'. Some of the vendors who claim to support cloud search actually mean that you can index content hosted on cloud-based servers, or that their software can be installed on a cloud drive, or their (often conventional) enterprise search can be installed on a remote (cloud) device. As always in search architecture, the devil is in the detail, not in the sales literature.

The hosted search model

Hosted search is the other common approach to off-premises services and services providing enterprise search. In the hosted model, you are responsible for configuring the search platform including defining the data sources to be indexed and searchable; creating the search forms; defining synonyms and perhaps best bets; and for running the reports you will use to understand your users' intent.

The software behind the scenes is simply a version of a standard commercial product. Hosted search often presents a simplified administrative interface, sometimes with fewer configuration options than the product licensed and installed 'on-prem'. While the management is less complex, the feature set for hosted search is generally rich and complete and is designed for a less technical staff to manage.

The contract

In the case of both cloud-based search and hosted search, it is important to look at the contract very closely. There can be restrictions on crawl and index frequency, and these services often include a seat-based or query volume licence, which may not be suitable for your organisation. It may also be difficult to add in third-party applications, such as search log analytics, to provide additional functionality.

It is not uncommon to find that contracts are based around the potential requirements of web site search and these may not be appropriate to enterprise search. One of the results of this web site focus is that not all of a document may be indexed, just perhaps the first 1500 characters. That might be adequate for a web page where the page construction places a substantial amount of text at the beginning of the page, but not for longer enterprise-style documents created in Microsoft Office.

The contractual problems can be especially difficult to resolve when an organisation already has an agreement with a SaaS vendor and now wants to add in a search application.

Implementation

In the same way 'Search as a service' is subtly different and perhaps easier to use than 'Cloud search', since the implementation of the various services can vary in complexity; and in the skills your team will need to implement search.

The HTML method

This approach requires creating HTML code on your web site that prompts the user for a query; sends the query to the hosted search service; and processes, formats, and displays the list of results in HTML format. For vendors who support simple HTML/Form integration, the implementation is relatively easy. You will need knowledge of, and skills in, HTML and HTML forms in order to add a search form to your site and to create a result page that will display the results as well as the ability to 'jump to' subsequent result pages.

The API approach

Some services require writing code using an API, short for 'Application Programming Interface'; this generally refers to using the vendor's REST API. Basically, this means the search form on your site programmatically sends users' requests to the server; and then processes and displays the structured list of results that are returned. The API approach is a bit more complex than the HTML method, but the results list is virtually identical.

Integrating with an API service may require more advanced capabilities. In addition to familiarity with HTML and forms, you may need to understand JSON, JavaScript or whichever scripting capability the vendor requires. If you have an outside consultant doing the implementation work, be sure to get the source code and documentation so you have the ability to modify the code at a later date independent of the initial implementation vendor.

Supported document types

Public facing web site content typically uses HTML and PDF formats. When it comes to internal web sites and content repositories, many additional formats are used; and the search service you select will need to support the formats in use in your organisation. This often includes content created with Microsoft Office, Open Office, RTF, PDF, Postscript and others.

Many of the SaaS services support those formats directly using open source or commercial 'filter' tools to accomplish the task of indexing and previewing your content. Others, like Algolia and Amazon CloudSearch, require content be converted into JSON, HTML or text prior to indexing, which can be a significant ongoing effort if your

internal content changes frequently. Verify that the service supports the document formats you use with the least effort. If your content changes rarely, conversion may not be a problem.

Security and personalisation

There are two other issues when approaching cloud and hosted search, and you'll need to understand them in order to select the best commercial solution for your organisation. Those issues are content security and personalisation.

Most companies that offer hosted search have optimised their products for public facing content, typically for public-facing websites or blogs. Almost by definition, methods of accessing content behind a firewall present issues for hosted search. Typically, solutions such as connecting via a VPM or providing a 'tunnel' through the corporate firewall are not acceptable to corporate security managers. As a result, enterprise search applications, which usually support LDAP or Active Directory, are not generally available to remote servers.

However, many of the search services do provide security; and depending on the vendor and your corporate security policies, you may find an acceptable solution. For example, Algolia lets you define an access control list, or ACL, for users who are authorised to access your indexed content; actually, viewing the document will likely only work for users who are inside the organisation or who have VPN access to your internal secure network. Swiftype, recently acquired by Elasticsearch, encrypts your content at rest and in transit, which may satisfy your organisation's security policy. But getting to the actual document may still be an issue.

Many other hosted search vendors support only the indexing of publicly accessible content. This is an area that is changing quickly, so check with the vendors you evaluate.

Other vulnerabilities

Even for those services that support security, there is an additional potential vulnerability.

Some hosted search sites include a feature that attempts to deliver results as the user types the query. For example, when a user types the letter 'a', results will appear that have words that start with the letter 'a'; and the search platform refines the search terms and results as the user types additional letters. The risk here is that the word list is typically maintained independently of the content index; and simply exposing the word to a user not authorised to view a document may be considered a security breach. For example, if a product under development should only be visible the development team, exposing the project name to other employees may be considered a breach.

Other examples of words that may accidentally breach security include 'promotions', 'layoffs', 'reorganisations', 'acquisitions', or 'mergers'. The very fact the word is suggested may provide information the person doing the query is not authorised to view.

An important issue to consider is the implications of the General Data Protection

Regulations of the European Union which come into force in May 2018. It is essential that the corporate legal team reviews the terms of contract against the provisions of GDPR.

Personalisation

Many of the hosted services integrate 'machine learning' technologies with their search platform. With sufficient activity, the search platform can 'learn' which documents are likely to be of interest for a given query, much like large web sites including Google and Amazon do.

Machine learning, or ML, is far more effective when users can be easily identified. This is most effective when users have unique login credentials or a fixed IP address. Of course, popular internet sites have large numbers of users and queries; and the more users and queries a site has, the better the ML technology can provide meaningful recommendations. As a result, machine learning may be far more effective for active e-commerce sites than for sites delivering content to a relatively small team of users.

Conclusions

Hosted search products provide high quality search results without the need to acquire and maintain expensive hardware and without the need for a large internal IT participation or even a large 'search team'. However, hosted search provides a somewhat limited set of capabilities which make it more suitable for public-facing content.

Nonetheless, a surprisingly large number of companies have entered the market, many with high quality, powerful technologies enhanced by easy to use interfaces. At this point with a relatively new capability, it's best to carefully identify your requirements so you can evaluate the various vendor services. And, because consolidation often takes place in markets like this, your due diligence should include a fallback option should the vendor you select merge with another company or exits the hosted search market.

Microsoft SharePoint and Office 365 search

Agnes Molnar

Microsoft SharePoint has a very special place in the enterprise search market. Traditionally, it has never been sold as an 'Enterprise Search' platform, rather as a platform for content services and business collaboration. There are other Search offerings provided by Microsoft, although these are not considered as Enterprise Search products: Bing is for Internet Search, Azure Search is a 'Search as a Service' (SaaS) offering, Office products have their own embedded Search functionalities, etc.

After acquiring the company FAST ESP in 2008, Microsoft kept its Enterprise Search platforms as separate products: Search Server and FAST Search for SharePoint. In those days, Microsoft was a solid player in Gartner's 'magic quadrant' for Enterprise Search. In 2013 Microsoft integrated FAST ESP and Search Server into SharePoint and Office 365 and no longer had stand-alone search offerings. Once the 'Search Server' name was dropped, Microsoft was no longer included in Gartner's 'Quadrant for Enterprise Search'. This was a sign to many that Microsoft was out of the search market.

However, with the latest version of SharePoint (2016) and Office 365 in the cloud, a new wave of innovation is using machine learning techniques to personalise and improve search and discovery. As a result, Microsoft is back in the Gartner Magic Quadrant for Insight Engines.

Connecting source systems to SharePoint search: content sources

One of the most important pillars of enterprise search is content. When considering and planning search, the first step is to define the systems it needs to be connected to. To establish a connection to a content management system, we have to configure Content Sources in SharePoint 2016.

In SharePoint 2016, the following types of content sources are available out-of-the-box:
- SharePoint sites (2016, 2013, and 2010)
- File shares
- Exchange public folders
- Websites
- BCS (Business connectivity services - databases and web services)

If a connector out-of-the-box is not available, content source systems can be connected to search by custom or third-party connector solutions. Integrating these external content sources into the search engine is critical to increasing employee productivity by providing better findability in a shorter time.

However, getting connected to these external systems is not as easy as it sounds. Out-of-the-box as well as custom connectors can connect to the particular source system usually by using standard APIs or direct database access. Custom connectors can be installed on the top of the SharePoint search APIs to connect them to the content source.

The connector is always attached to the SharePoint crawler and helps it to enumerate content from the system of origin. As soon as the crawler gets the documents, the process is the same as in the case of out-of-the-box connectors: the crawler sends the

content to the standard content processor for further processing. Once the content processing is done, the extracted information gets stored in the search index.

However, creating these connectors has several challenges. Writers need to know not only SharePoint but also the connected system's data and security models, APIs, database architecture, etc. Besides the data model, understanding the permission management is also critical, as many systems have different, non-active directory-based security models with custom permission features. These all have to be mapped to the users in SharePoint. Providing security trimmed results is a must.

Search in Office 365

Office 365 has a very special position on the market. Besides adding more collaboration features and capabilities, Microsoft also realised the importance of content findability. Microsoft has invested a lot into Microsoft Graph which stores people, content and their relationships in the cloud, and provides a new way of information discovery as the basis for many applications, including Microsoft Delve and the new personalised, 'modern' search.

Satya Nadella, CEO of Microsoft, talked about the new search capabilities and enhancements during his keynote at Microsoft Ignite, 2017. The vision is to create a personalised and behaviour-based search experience. The new, 'modern' search applications in Office 365 provide an easy-to-use interface to surface and discover recent content that should be relevant to the current user.

While it's really useful when the intent is to discover content, or get back to a recent document, this approach still provides very poor results when it comes to research, learning or aggregating content.

Another big limitation of this 'modern' experience is there is no way to configure or customise what and how the content is presented to the users.

At the same time, 'classic' search is still (and will continue to be) included in Office 365 and lots of configurations and customisations can be done there (see below). However, classic search has not been improved in recent years as Microsoft continues to focus all its search resources on the 'modern' experience.

Hybrid search

Microsoft's Cloud hybrid search, introduced in May 2016, has proven to be a solid and pragmatic solution. It's being used either as an on-ramp to the cloud or as a permanent strategy and is very effective for those already using Office 365.

Configuration and customisation options

When it comes to setting up, configuring and customising search in SharePoint and Office 365, there is no silver bullet action plan that works for everyone. However, there is a set of search components that have to be configured correctly.

- **Content sources** represent the source system connections as well as crawl schedule and other settings (SharePoint and hybrid only).
- **Search index** is the 'registry' of content that is the basis for all search features. The index is stored on-premises (SharePoint) or in the cloud (Office 365 and hybrid search).
- **Result sources** are subsets of the items stored in a search index. A result source can be used to provide pre-filtered results or create search verticals.
- **Query rules** can be used to modify how the queries are processed. By conditions

and actions, we can promote results (a.k.a. 'best bets'), to boost results, as well as to modify the ranking of search results.
- Both in SharePoint as well as Office 365 'classic' search, different types of results can be displayed in different ways. This behaviour can be achieved by defining **Result sources.**
- Everything that is displayed on the search UI or in a search-driven web part, is described by a **display template**. We have separate display templates for the result items, hover panels as well as refiners. Each display template can be customised, and we also can define our own by using HTML and JavaScript.

On the search user interface, we can find further elements that can be customised, such as the result set itself, highlighted result blocks (defined by query rules), the refinement panel or the hover panel (preview panel).

The skillset needed to implement search successfully in SharePoint, Office 365 or hybrid is broad. The infrastructure is complex, and the search team will need to cover a range of roles including for example a backend-engineer, a frontend-engineer, a relevancy engineer, a content curator, a designer and more.

Extending search

Treating search as a ready-to-go solution might seem to be the easiest option, but experience shows that organisations with the mindset of 'search as a platform' are much more satisfied with their search implementation.

Connectivity

To find any results in search, we have to get connected to the content sources. SharePoint 2013 offers several connectors out-of-the-box, for example, to SharePoint, file shares, Exchange public folders, or web sites. We can also get connected to simple databases and web services by using Business Connectivity Services (BCS).

This provides a good 'starting kit,' although real-world organisations always have other types of systems where content with high business value is stored. Integrating these external content sources into SharePoint search is critical to increasing employees' productivity by providing better findability in a shorter time.

However, getting connected to these external systems is not as easy as it sounds. Although SharePoint offers Business Connectivity Services, it is very limited in permission management and performance in enterprise scale. Although developing a custom connector to third-party systems is technically possible, it is highly recommended to consider purchasing an existing solution from a trusted vendor.

Classifying and unifying cross-system data

Getting connected and being able to 'pull in' the content is essential, although not enough. The next challenge is to classify and unify data coming from various systems. Classification is the part of content processing during which we put additional metadata on the document (in the search index) based on its existing characteristics or content. This requires extracting the content, transforming it by our pre-defined rules, and generating the new metadata.

Preparing and generating unified data with auto-classification solutions is evident, but the data coming from heterogeneous systems can be even more heterogeneous. The name of metadata fields, as well as the value sets, might be different, which results in a confusing user experience in the end. This is why we need to unify everything. Third-

party auto-classification tools can help with transforming the heterogeneous data into our unified standards.

User experience

To meet today's user expectations, the out-of-the-box user interface elements usually have to be extended: visual elements, charts, maps, diagrams, specific visual refiners, tables, hierarchical elements, etc. can all help the users achieve more by search.

Since most of these components are not available out-of-the-box, we have to write custom modules or purchase third party ones. In many cases, simple stylesheets or custom display templates can help a lot, too.

Skills needed

Many organisations may have the impression that search in SharePoint and Office 365 does not require any specific skills, believing that it's enough to turn it on for the magic to happen instantly. Most organisations still don't have specific roles with a responsibility for search. In most cases, search is considered part of the IT/SharePoint administrator role.

However, these companies miss a very important point. The out-of-the-box search experience can never fulfil the needs of any business. It needs server-side tuning (crawl and index optimisation, ranking customisation, metadata configuration, etc.) as well as UI customisations. Without a team with these skills, search will be only a 'default' feature that cannot support the organisation's specific requirements.

The future of search in SharePoint and Office 365

Microsoft's focus on 'cloud first' is reflected in its search developments. Microsoft has invested heavily in Office 365's new 'personalised' search with the false promise of offering a great search experience with zero investment.

There is no question that 'personalised' search provides relevant results when the user wants to get back to a recent document. But enterprise search is not always about the 'latest' document. It has to provide insights from an extended timeline, too. Unfortunately, these requirements cannot currently be met by Office 365. Organisations have to understand and analyse their requirements and decide when to use 'classic' search and when the new 'modern', 'personalised' Office 365 search experience is more relevant.

Recommended resources

https://searchexplained.com/is-sharepoint-search-dead/ by Jeff Fried, CTO of BA Insight
https://searchexplained.com/thoughts-modern-search-experiences-in-office365/ by Jeff Fried, CTO of BA Insight
https://searchexplained.com/modern-vs-classic-search-experiences-in-office-365/
https://searchexplained.com/thoughts-modern-search-experiences-in-office365/
https://searchexplained.com/infographic-search-components-to-configure-in-share-point-and-office-365-download-3yrh8v21elwinz6f/
https://searchexplained.com/sharepoint-search-configuration-process-infographic/
https://searchexplained.com/how-to-organize-content-sources-best-practices/
https://bainsight.com/blog/search-predictions-for-2018 by Jeff Fried, CTO of BA Insight

Search project planning

Martin White

It is not uncommon for search vendors to be quite vague about the schedule for implementing their technology. In practice it may take a least a year from the decision to begin work on defining requirements and selecting a vendor. Then there will be a period of technical and commercial due diligence before work can start on installing, testing and then releasing the new search application.

The table below sets out the main stages of a project to install a new search application. This table is based around commercial implementations. There is an important difference with open source solutions, which will often need far more custom development than commercial solutions. For example, user interfaces might need creating from scratch whereas a commercial solution will at least have a premade template UI. In total the time from the nominal decision to replace the application to full implementation will be around 18 months and could be even longer. As a result, the costs involved will fall into at least two financial years and that might be a challenge in gaining agreement to proceed with the project. It is not unknown for projects to come to premature halt when the full financial implications become visible even if the business case itself is a very sound one.

Vendors may well suggest that the project duration can be reduced because their software is so easy to install. This is especially the case with appliance vendors and with SaaS vendors. However most of the time is spent on ensuring that the requirements are clearly defined, and that the vendor can meet not only the current requirements but also those over a three-year business horizon, and the extent of the skills that the customer has in implementation. In general search software is not 'easy' to install if the search application is going to deliver significant value to the organisation across multiple repositories.

It is important to appreciate that in most cases there will be a requirement for implementation support even with commercial vendors. There are comparatively few systems integration companies that specialise in search integration, and so it may not be possible for the specialised companies to have much flexibility over the availability of expertise, especially if the implementation is large scale, multinational or requires a substantial amount of customisation.

The project schedule below also assumes that there is a search manager working full-time on the current implementation who has the experience to guide the selection process even if they do not manage the project themselves. The reality is that every project is different and this table can only be illustrative.

Pre-study - commercial	Initial appraisal of requirements, technical options and levels of investment. Consider the options between on-premise, hybrid and cloud solutions. Draw up list of vendors, search integrators and consultants. Visit other companies who have undertaken a similar project. Define stakeholders and project governance. Establish a project team. Scope out the user research. Undertake a content audit. Consider the requirements for metadata and taxonomy management.	1 month
Pre-study - open source	Draw up a list of open source core products, specialist open source integrators and consultants. Assess the benefits and challenges of building the complete application or using a semi-commercial framework based on open source components. Consider the options between on-premise, hybrid and cloud solutions. Assess in-house open source skills and experience, and whether these can be allocated to a major search project. Visit other companies who have undertaken a similar project. Define stakeholders and project governance. Establish a project team. Scope out the user research. Undertake a content audit. Consider the requirements for metadata and taxonomy management.	1 month
User requirements	Detailed survey of user requirements and business requirements over the next 2-3 years. Focus on the information needed to make good decisions, and the risks that could arise if these are not made on the best available information. Update/write a search strategy to provide context for the business case and the user requirements.	2 months

Prepare RFI	The Request for Information should cover both core functional requirements and commercial/implementation topics. Aim to keep it to no more than 30 pages; it will help the business and the team focus in on the core requirements. Define proofs-of-concept that may be required.	1 month
Vendor response	Allow time for the vendors to clarify issues in the RFI. Develop an evaluation methodology. Scoring the responses can ensure that every member of the team reads every response but in the case of search is not a definitive way of selecting a vendor. Assess the proposals.	2 months
Vendor selection	From the initial evaluation develop a list of perhaps three vendors with which to conduct more detailed discussions. Take into account the requirements for an integrator and undertake an appropriate level of due diligence on potential contractors. Agree a heads-of-agreement with the preferred vendor and integrator that includes an estimate of the Total Cost of Implementation over a 3-year period. Pricing models for search applications are highly complex and dependent on many variables. It will be difficult to compare the TOC between vendors.	2 months
Proof of concept (PoC)/due diligence	Set up and conduct Proof of Concept (PoC) tests on core requirements. Visit customers of the vendor.	2 months
Review period	Ensure that the outcomes of the work with the vendor are still aligned to user requirements, as these could have changed since the initial user research. There will not be a perfect match between user requirements and the functionality on offer, so be clear about the trade-offs that might need to be made.	1 month

Contract	A decision will need to be made about the scope of the contracts with the application vendor, the integration partner and any third-party software (such as a taxonomy management application), and yet all these contracts need to meet in the middle. At this stage it can be very valuable for a joint risk-assessment to be undertaken so that the respective roles in identifying and addressing project risks are clearly defined at the outset.	1 month
	Duration from initiation to contract.	12 months

This 12-month period is reasonably predictable with the exception of the Proof of Concept phase. Because search applications are development platforms, not products, there may will be a wider scope for the PoC phase than might be the case with other enterprise applications. The two-month period suggested in this table assumes that all the pre-planning on test collections and queries (for example) has been completed during the initial stages of the project,

However, the work that follows the contract decision is much more difficult to forecast and will be heavily dependent on the complexity and scale of the content repositories, and the degree of customisation and the technology basis (SaaS, appliance etc) of the solution. In terms of elapsed time these tasks will (at least to some degree) take place concurrently. The biggest gate is the first full crawl and the initial User Acceptance Testing (UAT) on customisation. It is not uncommon for this initial crawl to highlight a significant number of issues. Crawls take time. If the ingestion rate of one document a second a base case would be that one document a second works out at around 10 days for one million documents. There are many ways of reducing this crawl time but the more complex the procedures the more difficult it is to work out where something went wrong.

It is at this stage that meetings with customers of the vendor who have undertaken a broadly similar project are highly advisable. If there is no customer who fits into this category then the vendor, the implementation partner and your organisation will be learning as you go along. That does not mean the project will not be successful but does raise the overall project risk.

Installation pre-work	Undertake any changes and additions to the server architectures. Ensure that all security and disaster recovery requirements are identified and documented. Define the process for decommissioning the current search application. Consider the benefits of a pilot or Minimum Viable Product stage in the implementation. Define the Key Performance Indicators for the project and for the search application post-implementation. Establish the search team that is going to support the installation and then the implementation. Define the requirements and resources for usability testing. If an open source solution is chosen, there may be a need for significant custom development to be carried out both before and after installation of the core search technology.
Installation	Initial software loading and crawl/index runs. The initial crawl and index can be a lengthy process and often the process may have to be restarted once initial problems have been resolved.
Customisation	In reality installation and customisation will be undertaken in parallel, but for clarity they are set out here as two individual requirements.
UAT	Initial user acceptance testing.
Beta roll-out	Manage release of the search application to a control group of users.
Full implementation	Enterprise wide release.
Project assessment	Review of the status of the user experience after one month.

The elapsed time from contract agreement to enterprise roll-out could be as short as two months or as long as perhaps nine months. This means that for a large-scale multinational implementation it could be close to two years from initial vision to full implementation.

Content audit

David Hobbs

Effective content audits are a means of making decisions and understanding the impact of those decisions.

Carrying out a content audit should not be a mind-numbing activity. If you're finding it painfully boring, then you're probably doing it wrong. The key is to stay at the level of discussing rules about content rather than making decisions item-by-item. By looking at rules, you can see the impact of your decisions, like the effort that will be required over time to make the changes.

The scale of the content to be indexed, in terms of both volume and file format, will have a significant impact on both the license fees and professional services charges for search implementation. The volume of content will also have a direct impact on the duration of the initial crawl, and as a result the implementation schedule. File formats and specialised content (especially databases) will also determine the need for connectors. It is therefore very important to undertake a content audit at an early stage in the project. The audit will almost certainly prompt a discussion about how far back in time content needs to be indexed, and that is not an easy decision to make. Although an audit may have been carried out in the past, business requirements and the overall volume of content will probably have changed significantly over the period, and relying on a five-year old audit is not advisable.

There are seven rules of content audits
1. We want to make decisions
2. Not all content should be treated the same
3. Quality and effort are continuums
4. When deciding, we balance the resulting quality and effort to get there
5. We should make decisions based on rules
6. To make decisions, we need to explore our content
7. We need to understand the impact of our decisions

1. We want to make decisions

The only reason to do an audit is to make decisions. What are the decisions we want to make with respect to search?

- How can we clean/improve the content to be more effective for search?
- What content should be in and out of the scope of our search project?

Obviously poor content leads to poor search performance, especially for internal search since we lose some signals that external search engines can use. Our primary concern should be in improving content. Here are some ways we might improve our content for search:

- Deleting content
- Rewriting content to align with common searches
- Reorganising sections of content
- Changing the format of a piece of content
- Technical improvements
- Better structuring content to allow more structured search results
- Better integration of orphaned pages

The above are dispositions for improving the content. There are also decisions to:

- Keep the content as is (which can include things like archiving in some way)
- Delete content (which is often one of the best ways to improve overall content quality)

For more on dispositions see http://hobbs.direct/dispositions.

2. Not all content should be treated the same

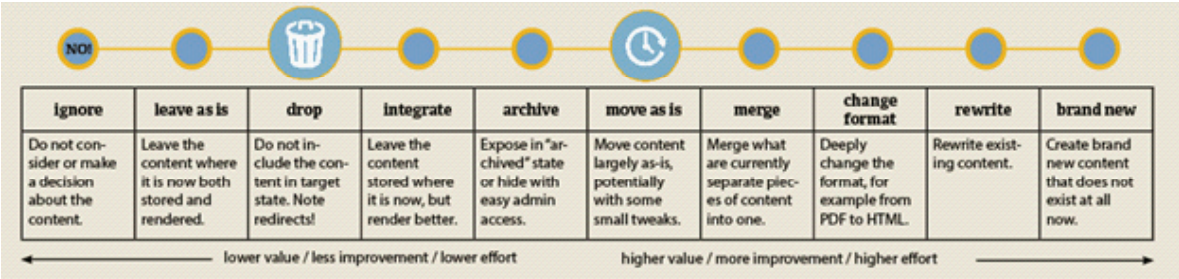
One of the most common mistakes is to make very blunt, global decisions about content. And the decision is usually between just forklifting the content or deleting it. But there are reasons why some content would be treated differently:

- The relative importance of the content
- The popularity of the content
- The current performance of the content
- The current quality of the content
- The importance of the content to achieving the goals of the larger search improvement effort.

For instance, if you are improving the usefulness of the top search results for a list of key topics, then it is worth spending more effort to improve the pages related to those topics than other pages.

3. Quality and effort are continuums

The dispositions are how we will treat the content, and these lie on a continuum of effort level:



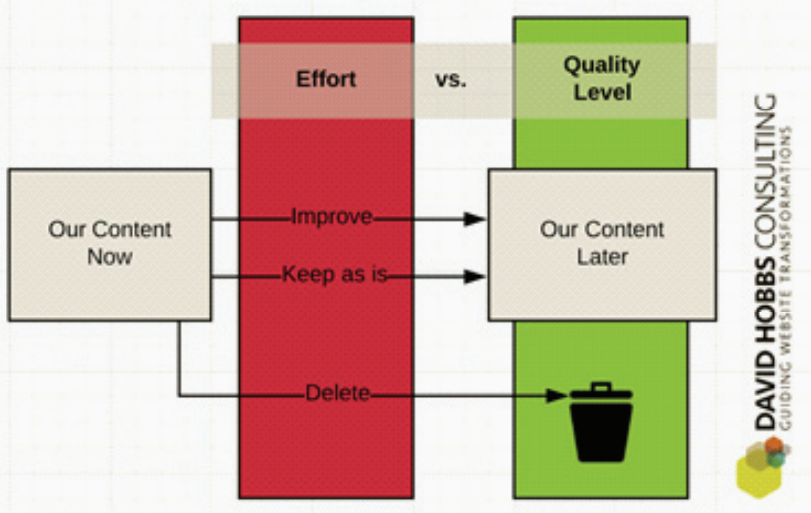
But this is just the mechanics of how we are getting from point A to point B.

We are also tweaking the resulting quality. This is sometimes correlated with effort but sometimes not. For instance, dropping content may be relatively straightforward but the positive impact quite high. Similarly, you may be able to better integrate with an existing repository of content rather than move it entirely to a new system.

Quality is certainly multi-dimensional, from purely technical to more editorial, so not just a linear progression. But it certainly is not just a high quality / low quality switch. As we go through different slices of content we want to consider what does that content hope to achieve, how important is the content, and how we could increase the quality of the content to better achieve the goals.

4. When deciding, we balance the resulting quality and effort to get there

We don't have infinite time or limitless resources. Our decisions are bound by constraints. One of the advantages of doing a methodical content analysis is that we can optimise our effort and quality rather than simply throwing our efforts to the wind and hoping for the best.



This means that our decisions need to be:
- Iterative. We need to dance between projected effort and quality level, and our first stab at decisions may result in too low quality or too much effort
- Creative. As mentioned above, quality and effort are on continuums. This means we have the chance to be creative about our approach to handling content

5. We should make decisions based on *rules*

There are two essential truths about making content decisions:
- The obvious. We need to make a decision about every piece of content
- The subtle. We do not need to look at every piece of content to make that decision

To pick an easy example, let's say we have a thousand documents of type "meeting notes." Perhaps the most common approaches would be: a) delete them all, b) keep/move them all as-is, or c) let the owners decide what to do with it. The first two are based on rules, but the last one (perhaps the most common?) is the most dangerous since there's a good chance the search experience will not improve after the content work is done. We can be more refined than that. For example, we could do things like "delete all meeting notes over a year old except board meeting notes which we keep for six years".

In other words, we can use rules to make decisions. Key advantages of rules include: 1) consistency across the enterprise rather than inconsistency due to ad hoc, localised decisions by division, group, or team, and 2) discussions more at the level of business needs rather than horse trading specific content.

Rules can define three things:

1. bucket (a swathe of content that will be treated the same)
2. disposition (the treatment, as discussed above)
3. assignment (who does the work)

We take information about our content (for instance, in the example below Folder1, Folder2, and subdomain) in order to define the rules in tabular form:

Folder1	Folder2	subdomain	bucket	disposition	assignment
blog	*	www	blog articles	automatically move as is	integrator
media	press-releases	www	press releases	drop with deep redirect	editors
media	*	www	other media	manually move as is	editors
*		store	product pages	rewrite	editors

6. To make decisions, we need to explore our content

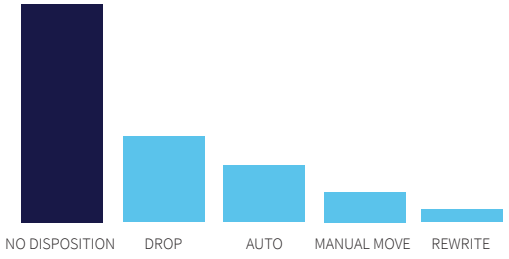
Although broadly we know what our goals are in conducting a content audit, our efforts are really an exploration of the content. In particular, we don't know in advance what information we need to make our decisions.

For instance, if we generate our content list from a spidering tool we will probably get fields like URL, title, H1 tag, and meta description. We may need to enhance this information in two ways:

- By adding information. When we add information, we get information from another source. For instance, we may wish to weave in content usage data that comes from an analytics package.
- By extracting information. Sometimes useful information is right in front of us but not in a useful format for making decisions. The URL http://widetsdivision.intranet.happycompany.com/content/meeting_notes/team_meeting_to_discuss_tps_report_18Jan1995.pdf on its own may not be very useful. But we can extract useful information like:
  - The "folder" (like folder 2 = "meeting_notes")
  - The type of file (for instance, we might take the extension of pdf, lump that in with other similar document types like MS Word, and say it's a "Structured Long-Form Document".
  - The fact that the subdomain is "widgetsdivision" may be useful.

At each step of our audit, we need to decide whether it is worth getting the more refined information that the current step has uncovered. Sometimes a method of extracting more information is to actually pull out elements of the document itself, like scraping out whether there is a table on pages by looking at the HTML.

To help work through our content, we can group and graph actual content counts that are in each disposition.



7. We need to understand the impact of our decisions

There are several impacts of our decisions that we need to consider:

- Confirm the rules make sense. Sometimes we define rules that upon inspection require further refinement. For example, we may define all blog posts as those that are on the blog subdomain, but upon further consideration realise that there are multiple blogs, each with its own pattern. One of the most effective ways of validating the rules is to randomly sample different buckets of content. For instance, the chart above could be clickable, so that when you click on the "drop" column you see those that will be dropped. Since the list is probably large, you can then randomly sample the members of that group.
- Estimate the effort of making the change. To estimate the effort, you can first make a stab at the effort of moving one piece of content for each type, then multiply that out by how many there are. See http://hobbs.direct/estimate for more
- Determine the resourcing impacts. Depending on the size of your project, there may be a wide range of people and/or organisations involved in improving your content. By assigning to different people/teams, you can see if the division of labour is possible to execute
- Define what the sequencing will be. It's easy to wave our hands and declare that one region/site/site section will be improved before another, but quite another to see the how the effort and resourcing works out

Remember: decisions and rules!

We really only want to be doing audits to make decisions. The best way to make decisions is by rules. These rules will define what to do with every piece of content, but the decision of what to do does not need to be made by inspecting each piece of content.

Project budget

Miles Kehoe and Charlie Hull

A structure for a search budget

Enterprise search vendors are very reluctant to give any indication of the likely budget needed to implement their software. There are a few exceptions, notably Mindbreeze (which is a search appliance) and dtSearch, which has always had a very transparent pricing structure. What is virtually certain is that every vendor will have a slightly different way of presenting the costs associated with implementation so direct comparison is going to be a challenge.

If open source software is chosen, then there is no vendor. This removes some cost elements and you might reasonably expect that the overall cost will be lower. However, much more software development may be required, raising the cost of professional services and also raising internal costs. Beware of the misconception that open source software is 'free' – it is 'freely available' but adopting it is not cost-free.

The main cost elements are

- Vendor software licenses
- Vendor professional fees
- Vendor maintenance and support fees
- Third-party software license fees
- Professional services from an implementation partner
- Internal costs for hardware, software and the support team
- Proof of concept/validation charges

Vendor software licenses

Vendor software licenses are almost always volume related. The volume elements can be the number of documents/content items, the number of users or file storage capacity. In the worst case each entry in a database and/or each individual email may be represented as a separate content item in the search engine, so these numbers can be very large. To arrive at a figure for the number of documents requires a content audit because the IT department may well know the amount of storage but not the number of files. The format of the files will also be important (see chapter on Content audit, page 42). The number probably needs to be 'correct' to the nearest million. Arriving at a static current number is going to take time but extrapolating over the next 3-5 years may be more difficult. Many of the files will be versions or copies but these all count as a content item when it comes to indexing.

Another factor is how far back the search application should go into archive material, where file formats may not be standard Microsoft Office and metadata tagging is likely to be partial or non-existent.

The number of users may seem to be quite straightforward until a vendor asks you to categorise types of user: broadly speaking users who are undertaking a simple Google-like search and those who are going to spend a considerable amount of time at each search session. Another issue is that many companies may have a substantial number of employees in manufacturing or out on site as service and support staff. Their use of search may be very limited, but they will still expect access when the need arises. Any such categorisation is very difficult to define, agree and monitor.

The time period for licenses will also be open to discussion. Many vendors offer a perpetual license (which has no expiry date) or an enterprise license (which requires little or no clarity on the number of content items or the number of users). These and other variants require the customer to have considerable clarity about the future which is close to crystal-ball gazing. Any company that gets to the stage of negotiating a contract without a well-prepared three-to-five-year search strategy is either going to pay too much or buy too little.

Vendor professional fees

As with any enterprise software application the professional fees will almost certainly be substantially more than the software costs. Installing and implementing search software is a highly skilled business and the engineers working for the vendor will be paid well because the last thing that the vendor wants is for a skilled engineer to leave. Replacing them can be time-consuming and there is a substantial amount of training to be given. Gaining clarity on the roles that the professional services team will play is very important. Some elements of the service provision may be fixed fee, but it is more normal to be charged on a time and materials basis. There could also be a substantial travel and subsistence costs if engineers need to be flown in from the USA to Europe, or any other long-haul journey. These costs (and the same for implementation partner costs) will inevitably be front loaded.

Vendor maintenance and support fees

Maintenance and support fees will often be presented as a series of packages with an increasing level of support. Often there is a lot of attention paid to '24/7' support with response times of a few hours. This usually refers to being able to register a ticket with a human being and not the time taken to solve the problem. Few vendors are willing to commit to a resolution timescale. There is no reason why a prospective purchaser cannot request a customised service package which integrates well with the level of service that can be provided by a well-trained IT team.

Third-party license fees

Search applications are very modular, and there could be a requirement for specific applications to meet a customer requirement which is best met by using a product from a third-party vendor. Examples would include high-end analytics visualisation, specialist connectors and linguistic products for non-European languages.

Professional services from an implementation partner

Relatively few search software vendors will undertake all elements of an implementation. Their business is in developing and selling software, not professional services. Almost certainly there will be a requirement for additional support, perhaps because the implementation is global, but the vendor operates only in the USA or Europe. This support could come from an existing implementation partner or more likely would come from what is a very small number of specialist search implementation companies that have partnership agreements with one or more vendors. As is the case with vendor professional services staff, partner engineers are also in short supply so day rates of €2500 are quite usual.

The involvement of an implementation partner raises the issue as to whether there are separate contracts with the vendor and the partner, or a contract with just the partner, who then supplies and implements the software.

Again, these professional services fees will almost certainly be on a time and materials basis, and that makes budget forecasting by the customer very difficult.

Internal costs for hardware, software and the support team

Enterprise search applications are development platforms and not products, so a development + production environment will be the norm. Network bandwidth can be an issue – users expect response times of no more than 500 milliseconds wherever they are on planet Earth, a requirement that few other enterprise applications will need to match outside of e-commerce applications. There could well need to be additional hardware and associated operating licenses to be procured and additional network capacity bought in for core global operations in (as a good example) China.

To a significant extent the quality of search performance is a function of the investment in search support staff. The team costs (which exclude IT staff!) may not need to be taken into account but if they are, the percentage addition on the cost could be considerable. For companies of between 5000 and 10,000 search users (i.e. excluding manufacturing) you can probably get away with three full-time staff on the search team, even if not co-located. Above 10,000 users then you may need at least four. An experienced enterprise search manager can command a salary of at least €120k, and so a team of four could be at the €500k salary level, or €1m if overhead allocations are taken into account. Recruitment costs may also need to be budgeted for.

Proof of concept/validation costs

Some vendors will make a charge for the usually two-month long proof of concept phase but may also set this against the costs for the implementation. Others will not make a charge or will ask for a fixed price fee to cover administrative and related costs.

How much?

Because of the number of elements in the cost structure it is very difficult to give a good answer to this question. If you have 10,000 users (or more) and perhaps 30 million files to index, then the external costs from the vendor and implementation partner could be of the order of €2-3 million over a three-year period. However perhaps 60% or more will be professional services costs, and the scales of these will not be obvious until well into the contract negotiation. As with all software and services contracts everything is open to negotiation. Factors that could bring down the price include offering to act as a reference site, investing in a wider range of skills in the project team and offering the vendor an opportunity to gain a blue-chip client in a market sector that is core to their business development.

How long?

Contract negotiation will almost certainly bring up the issue of the respective scopes of the vendor and the implementor, as well as subsidiary contracts with third-party software suppliers. It is advisable to develop a draft structure to the contracts at an early stage, often referred to as a Heads of Agreement. Even then expect the negotiations to take at least two months from the Heads of Agreement to the Signed Contract. There is an immense amount of detail to get through and your Procurement Department will probably never have worked with the vendor before, so there are no established lines of contact and trust.

The future of search

Martin White

"Electronic Digital Machines for High Speed Information Searching" was the title of a PhD thesis written by Phillip Bagley in 1951 and set out a vision that came to fruition in the early 1960s. By 1964 natural language question answering applications were being used by the Los Angeles Police Department. Progress over the last 50 years has been steady, with many of the academic research advances taking some time to be integrated into commercial applications. Probably the most visible development has been the introduction of facets into the search interface. The approach was initially developed at the University of Huddersfield (UK) by Dr Stephen Pollitt in the mid-1990s and then further developed by Professor Marti Hearst.

The amount of research being undertaken into improving the performance of search applications increases year by year. Over the last 40 years of ACM SIGIR conferences nearly 5000 papers have been presented. A search on Google Scholar for research papers on information retrieval indicates there are over 1,700,000 papers, with almost 90,000 published in the last five years. This is of course largely academic research which may take some time to emerge into commercial products. In addition, commercial search vendors and the open source community are engaged on development projects driven by user requirements.

In 2017 there was a great deal of publicity from vendors about the benefits of 'cognitive search' (also referred to as 'insight engines') and the benefits of incorporating artificial intelligence (AI) and machine learning into search applications. The value of using AI techniques in search was recognised in the mid-1970s but it is only over the last few years that improvements in machine processing and in efficient algorithm design have brought these two technologies into commercial products.

One of the promised benefits of AI and machine learning is the ability to present sets of results, and indeed extracted information, to users that is highly personalised to their work environment. This is seen as a way of reducing the effort to look through a long list of results, of which only a small percentage are of direct interest to the search user. In principle these would seem to be substantial benefits, but the issue is whether personalisation will inhibit innovation and learning by restricting the scope of the search, possibly in ways that the employee is not aware of.

Another approach to information discovery is the use of digital assistants, or chat bots. These are now being widely used in the consumer sector and in principle there is no reason why the technology could not be used within the enterprise. The key question is whether the nature of enterprise queries might be a challenge for the technology, at least at the present moment. Research on understanding the differences between typed and spoken queries dates back to at least 2007. It seems that for short queries there are few problems but with longer queries there may well be a requirement for reformulating the query. Technically the quality will improve; whether employees will be willing for their queries and the responses to be overheard is another issue.

Despite these advances there are a number of areas where progress in search technology seems to have stalled. Organisations now recognise the value of team working, and there are many (perhaps too many!) enterprise social networking and collaboration applications. However, in the case of search, the applications remain one person

– one screen. In the 1970s the benefits of using a search expert together with a topic expert were very clear. As search became more intuitive the end user was increasingly capable of carrying out the search themselves. In the case of teams, this means that either each member of a team undertakes their own research, or one person undertakes the research on behalf of the team. This seems counter to the benefits of sharing knowledge within a team. Although there is quite a substantial amount of recent research that indicates the benefits of collaborative information seeking (CIS) there are no stand-alone applications and none of the current collaboration tools support CIS.

There seems to be little progress in developing search interfaces that can be customised by the user. In the early days of enterprise information portals one of the benefits these portals offered was functionality to drag and drop a personal selection of applications onto a desktop. There would seem to be value in being able to personalise search user interfaces through dragging and dropping facets and other features to create a personalised work space. This could be of particular importance in federated search systems where the user wishes to maintain native access to the UIs of each system, rather than use a generic UI. An example of this is the experimental PerFedPat project. However, this approach has not been adopted by commercial vendors.

Many vendors are creating the impression that keyword search is dead and that enterprise search is dead. The problem they are promising to solve is akin to mind-reading with a computer application. For example, what signals can be used to optimise the results from a search being undertaken by an employee when they have entered [HROnline] as a query term? What users may be looking for is the application that was called HROnline but it now called PerfectPerson. That is a relatively easy problem to solve. What is more difficult it to know whether they are looking for information about the application, the person responsible for the application, or the log-on screen to the application.

This is where it is important to be aware of differences between internal search and external public web search. Web sites are full of information that the people running the web site want to be found, and they will make considerable effort to ensure that the content is presented in a way that applications like Google and Bing can find. The web site itself will usually be quite focused in its content and it will be supported by a team looking anxiously at the search logs. What is unknown is the identity of the user and any means of contacting them direct.

Enterprise content is very unfocused. Even the breadth and volume of information on an intranet might be counted in millions of documents contributed by hundreds of employees on tens of different topics. They are not writing the content to be found by a search application and rarely will there be any attempt to measure user satisfaction with the search application even though the identity of every user is known. Techniques which work well with web site and e-commerce applications may not be as effective on enterprise content.

What advice can we offer you when responding to a vendor offering novel technology?

- Ask what percentage of their customer base has upgraded to the current version in the course of the previous twelve months.
- Ask to be introduced to a couple of customers using the current application and set up a face-to-face meeting where the application can be demonstrated and its functionality assessed.

- Ask what skills you will need to have available in the validation phase, the implementation phase and then on into the production version
- Ask what changes you might have to make to processes and training to optimise the benefits from the novel technology.
- Ask for a report to be prepared by the vendor into exactly how its technology is going to make a difference to the search experience of some or all of your employees.
- Ask what the price differential is of the enhanced technology version against the prior version and consider whether the increase represents value for money.
- Ask what else is in the pipeline for the next year, because by the time you have been through the procurement process and begun implementation it will be next year.

You should be looking for good answers for all seven questions. If you get them then no one will be more pleased than the contributors to this report because your organisation will be in a strong position to enhance business performance and reduce business risk.

That is what all search applications should be delivering.

Critical success factors

The Search Network

This list of twelve critical success factors is based on the collective experience of the authors of this report ('the Search Network') working on search implementation projects in the USA and in Europe. They are not in any particular order of importance.

1. Content quality is essential for quality search

Good search technology will quickly reveal poor content. There should be guidelines for content and metadata quality. It is of little benefit to the organisation if a search lists twenty algorithmically relevant documents with a content quality that renders them unfit to be trusted, used and shared.

2. Invest in a search support team

A search support team with skills, enthusiasm, organisational knowledge and networks is essential to the success of an enterprise search project. Search applications need a blend of business, technology and information retrieval skills to get the best results.

3. Get the best out of the current investment in search

There is usually much that can be done to improve the current search applications once the search team and the search vendor work together on options and priorities. The information gained from search log files needs to be integrated with the outcomes of surveys, focus groups and interviews if user requirements are to be clearly established.

4. Recognise that enterprise search is an approach and not a technology

Enterprise search is about creating a managed search environment that ensures employees find the information they need to achieve organisational and/or personal objectives. Even in smaller organisations there may well be a case for more than one search application, and of course many enterprise applications will have embedded search. All these applications need to be integrated into an enterprise search strategy.

5. Set search within an information management strategy

A search strategy needs to be grounded in an organisational commitment to information management which recognises that information is a business asset. The information management strategy should define topics such as content quality requirements, language policies and document security policies, all of which have an impact on search performance.

6. Understand user requirements and monitor user satisfaction

Search logs will indicate the queries that have been used but not why the information was being sought. It is important to understand the business and information context of users and to monitor user satisfaction with search. Developing personas and use cases/tasks is especially valuable.

7. Recognise that information discovery involves searching, browsing and monitoring

Users need to be able to search when needed, browse when needed and monitor as needed. These three processes need to be linked together to provide an effective information discovery environment. This is especially the case with intranet search applications where the transitions between browse and search need to be as seamless as possible.

8. Assess the business impact of search

Go beyond search log analysis and user satisfaction surveys and understand where search is making an impact on business performance. Document and highlight where search has made a positive impact on organisational performance and do not be afraid to describe search failures. Much can be learned from mistakes.

9. Train and support your users

Search is not intuitive. It is far more than entering words into the search query box. Make sure that there is a range of on-line and face-to-face advice available. The process of training will highlight areas for improvement for other users.

10. Remember that search is a dialogue

Users will have complex and often ill-defined queries that require them to be able to refine their query and re-evaluate the results with the minimum of effort. The options for refinement (especially facets and filters) can make the user interface very complicated. Usability testing is essential in enhancing the query management process

11. Do not rush the implementation process

Because search applications are development platforms and not products the process of optimisation takes a combination of high-quality user requirements research and well-designed testing prior to launch. Most organisations will not have recent experience of search implementation so learning from other organisations can be very valuable.

12. Regard achieving search excellence as a journey, and not a project

The process of ensuring that search is meeting user requirements never comes to an end. Every day there are new employees, new business challenges, new business opportunities, and new developments in search technology. Search should never be a ‘project’ but instead be a way of working.

Appendix A Enterprise search software

The Search Network

This list is included only to provide a starting point in creating a shortlist for an enterprise search project. There is no implied endorsement by members of the Search Network. A more comprehensive list, including many specialist software vendors, can be found at http://www.enterprisesearchbook.com/vendors/vendors-directory/

A list of some search integration companies can be found at http://www.enterprisesearchbook.com/vendors/implementers/

Company	HQ	Category	Gartner 2017	Forrester 2017
Algolia	USA	SaaS		
Antidot	France	Commercial		
Amazon	USA	SaaS		
Attivio	USA	Commercial	Yes	Yes
Autonomy	UK	Commercial	Yes	Yes
BAInsight	USA	Commercial		
Coveo	USA	Commercial	Yes	Yes
dTSearch	USA	Commercial		
Elastic	Netherlands	Open Source		Yes
Exalead	France	Commercial	Yes	
Findwise	Sweden	Open Source		
Flax	UK	Open Source		
France Labs	France	Open Source		
Funnelback	N/A	Commercial	Yes	
IBM Watson	USA	Commercial	Yes	Yes
IntraFind	Germany	Open Source		
Lucene	N/A	Open Source		
Lucidworks	USA	Open Source	Yes	
Mindbreeze	Austria	Appliance	Yes	
Microsoft SharePoint	USA	Commercial	Yes	
Microsoft Azure	USA	SaaS		
Open Source Connections	USA	Open Source		
OpenText	Canada	Commercial		
Oracle Secure Search	USA	Commercial		
RAVN	UK	Commercial		Yes
Searchblox	USA	SaaS		
Searchify	USA	SaaS		
Sinequa	France	Commercial	Yes	Yes
Squirro	Switzerland	Commercial		Yes
Solr	N/A	Open Source		
Swiftype	USA	SaaS		
Vespa	USA	Open Source		

Notes

Findwise, France Labs, IntraFind and Lucidworks are based around Lucene, Solr and in the case of IntraFind, Elasticsearch. However, these companies also integrate modules which are provided on a commercial basis, and so in effect are a hybrid of open source and commercial products.

IBM, Microsoft and Oracle do not offer stand-alone search applications.

In March 2018 Elastic, the company behind Elasticsearch, announced it was 'opening up the code' for its X-Pack, various additions to core Elasticsearch available under a commercial license. It should be noted that this 'open' code does not meet the official definition of 'open source' (which Elastic has admitted) as the license to be used will be written by Elastic. At the time of writing the license has not been made available. Users and contributors to the code may still have to pay to use some or all of the X-Pack features, and the exact level of 'openness' is still unclear and may be confusing to end users.

Appendix B Search strategy checklist A-Z

Martin White

The objective of this checklist is to ensure that all the topics that should be covered in a strategy or in a statement of requirements are considered. The Status column provides an indication of how much of the required information to complete this topic in a search strategy is currently available.

A - We have all the required information to hand and can write this section with no additional research.
B - The information is available internally, and it would be fairly easy to collect it
C - There is very little information available internally and it would take some time to collect and collate it into a document.
D - This topic is not relevant to the current project.

Topic	Status
1. Accessibility	
Sets out the extent to which the search applications meet the Web Accessibility Initiative WCAG Guidelines to an acceptable level	
2. Acquisition	
The extent to which the search applications could be extended in an acquisition or merger situation	
3. Architecture	
Server and network architecture requirements and server availability	
4. Best bets	
The user requirements for best bets and how they will be reviewed and revised	
5. Big Data	
Integration of the search and Big Data strategies, especially around common metadata schemas	
6. Budget	
License costs, vendor maintenance and support, and staff costs	
7. Business cases	
Summary of the evidence from user requirements research to support and prioritise specific business cases, including the potential business impact	
8. Cloud search	
The potential benefits and challenges from implementing cloud (or hybrid) search	
9. Communications	
The communications strategy and forward communications program for stakeholders and users	
10. Connectors	
Requirements for connectors and associated support from suppliers	

Topic	Status
11. Content analytics	
The extent to which the organisation will benefit from implementing content analytics solutions and the relationship of these solutions to search	
12. Content quality	
Requirements for content quality and content curation to enhance search performance, ideally placed within an information life cycle framework	
13. Content scope	
A list of the content being crawled and indexed	
14. Crawl management	
Optimal crawl schedules to balance user requirements with any architecture/performance constraints	
15. Dependencies	
Business or technical dependencies that could impact search performance and search satisfaction	
16. Development plan	
The opportunities for enhancing the search environment over the following two years, based on user requirements research and business objectives matched against resources	
17. Disaster recovery	
Disaster recovery plans with Recovery Time Objective (RTO) and Recovery Point Objective (RPO) requirements	
18. eDiscovery	
If appropriate, the touch points between the eDiscovery strategy and the search strategy, especially regarding the sharing of skills	
19. Expertise search	
Linking the requirements for expertise search from a knowledge management strategy with the search strategy	
20. External search	
The requirements for search access to external information resources on e.g.research, competitors, and market opportunities	
21. Federated search	
Current and potential opportunities and challenges for implementing federated search	
22. Feedback	
How users will be able to feedback comments and suggestions to the search team	
23. Governance	
The ownership of the search budget and search strategy, together with roles, responsibilities, and lines of reporting for members of the search team	
24. Help desk management	
The relationship between the IT Help Desk team and ticket system and the search help desk	

Topic	Status
25. Information management	
A summary of the organisation's information management strategy with particular reference to the requirements and objectives for the search strategy	
26. IT liaison	
Service-level agreements with IT departments for support and development, including the requirement for staff with specific skills to be available	
27. Key performance indicators	
Definition of a set of periodic key performance indicators that relate to the business cases and business impact requirement	
28. Language	
Setting out any requirements for indexing and searching in languages other than the nominal corporate language.	
29. Legal conformance	
Requirements to conform to data privacy, Freedom of Information, and export license controls	
30. Licenses	
List of licenses by vendor and license renewal date so that the implications of a merger or acquisition of the vendor can be quickly assessed	
31. Metadata	
A summary of metadata schema, controlled term lists, thesauri, and relevant master data schema	
32. Metrics	
A summary of the suite of performance, discovery, satisfaction, and impact metrics, together with required benchmark levels	
33. Migration	
The implications for search as an element in a content migration strategy	
34. Mobile	
How the search applications will be implemented on mobile devices, together with an assessment of the need for cross-device support	
35. Open source	
Sets out the organisation's approach to using open source applications	
36. People search	
The requirements for people search	
37. Performance	
The technical (network/server) performance benchmarks for crawl, index, query, and result display	
38. Risk register	
A risk analysis relating both to operational and strategic risks for the search application, and the consequential risks to the organisation	

Topic	Status
39. Roadmap	
Release dates for upgrades to search applications, the basis on which they would be implemented, and development roadmaps for other enterprise applications	
40. Scope	
Confirmation of the repositories to be crawled and indexed in order to meet user and business requirements, the search applications to be included in the strategy, and the search applications that are being excluded	
41 Search based applications and bots	
Set out policies for the development, testing and on-going evaluation of SBA and bots	
42. Search support team	
Operational responsibilities and reporting lines for the search support team, including requirements for training	
43. Security	
Summary of security requirements covering confidentiality, integrity, and availability in line with ISO 27001 and with internal document circulation policies	
44. SharePoint strategy	
As appropriate, sets out the scope of Microsoft SharePoint adoption and development with particular attention to hybrid and cloud implementation	
45. Stakeholders	
Confirmation of the stakeholders and other members of the search community, using the RACI model	
46. User training	
Provision of training courses for search users, especially new joiners and staff in search-intensive roles	
47. Usability tests	
The scope and schedule for on-site and remote usability testing	
48. User requirements	
Defines the core user requirements as personas and use cases	
49. User interface	
Sets out any proposed changes to the user interface to meet user requirements, including the development, testing, and implementation schedules	
50. Website search	
Sets out the management and operational links between internal and external search	

Search resources: books and blogs

A reasonably complete list of books on information retrieval and search can be found on the Enterprise Search Book site. The books listed below represent a core library which should be on the bookshelf of any manager with enterprise search responsibilities.

Books

The Inquiring Organisation

Chun Wei Choo, 2015. Oxford University Press. (Review)

The importance of this book is that it provides a context for search within an overall integration of the value of information and knowledge to the organisation.

Introduction to Information Behaviour

Nigel Ford, 2015. Facet Publishing. (Review)

Information seeking models are a special case of information behaviours. They form the basis of use cases for search, and the design of user interfaces.

Designing the Search Experience

Tony Russell-Rose and Tyler Tate, 2012. (book website) (Review)

This book takes a deeper look into information seeking models, using them to consider how best to design user interfaces.

Enterprise Search

Martin White, 2nd Edition 2015. O'Reilly Media (book website)

A book for search managers without a technical background that supports the entire process from building a business case through to evaluating performance.

Searching the Enterprise

Udo Kruschwitz and Charlie Hull, 2017. Now Publishers (Review)

The authors provide an important bridge between information retrieval research and the practical implementation of search applications.

Relevant Search

Doug Turnbull and John Berryman, 2015. Manning Publications. (book website) (Review)

The objective of all search applications is to deliver the most relevant results as early as possible in the list of results. Although based around the management of Lucene and Solr this book is applicable to any search application.

Search Analytics for your Site

Louis Rosenfeld, 2011. Rosenfeld Media (Review)

This introduction to search analytics is primarily about websites and intranets but the principles apply to enterprise search.

Text Data Management and Analysis

ChengXiang Zhai and Sean Massung, 2016. ACM/Morgan&Claypool (Review)

A very comprehensive handbook on the technology of information retrieval and content analytics based on a highly regarded MOOC.

This is a list of blogs whose authors comment on aspects of search technology and implementation on a reasonably frequent basis.

Blogs

All About Search Ronald Baan
Beyond Search Stephen Arnold
Breakthrough Analysis Seth Grimes
Complex Discovery Rob Robinson
Concept Searching Corporate blog
Coveo Insights Corporate Blog
Daniel Tunkelang
Data Dexterity Corporate blog for Attivio
Do More With Search BA Insight corporate blog
Elastic Corporate blog
Enterprise Search Miles Kehoe
Exalead Corporate blog
Flax Charlie Hull
Funnelback Corporate blog
Information Interaction Tony Russell-Rose
Intranet Focus Martin White
LucidWorks Corporate blog
Matt McDermott
Opensource Connections Corporate blog
Searchblox Corporate blog
Search Chronicles Paul Nelson, Search Technologies
Search Explained Agnes Molnar
Sinequa Corporate blog
State of Enterprise Search Edwin Stauthamer (in English and Dutch)
Synaptica Corporate blog
Systems Thinking Paul Cleverley
Tech and Me Mikael Svenson

Glossary

The Search Network

Absolute boosting
Ensuring that a specified document always appears at the same point in a results set, or always appears on the first page of results.

Access control list (ACL)
Defines permissions to access a specific repository, a set of documents, or a section of a document.

Advanced search
The provision of a search user interface which prompts the user to enter additional terms to assist in ranking results, often using Boolean operators.

Apache
The non-profit Apache Foundation provides support for a wide range of open source projects, including Lucene and Solr.

Appliance
A search application pre-installed on a server ready for insertion into a standard server rack.

Auto-categorisation
An automated process for creating a classification system (or taxonomy) from a collection of nominally related documents.

Auto-classification
An automated process for assigning metadata or index values to documents, usually in conjunction with an existing taxonomy.

Average response time
An average of the time taken for the search engine to respond to a query, or the average end-to-end time of a query.

Best bets
Results that are selected to appear at the top of a list of results that provide a context for other documents generated and ranked by the search application.

BM25
A ranking function developed in the 1990s but still widely used. It has its origins in the tf.idf ranking function.

Boolean operators
A widely used approach to create search queries; examples include And, OR, and NOT—for example, information AND management.

Boolean search
A search query using Boolean operators.

Boosting
Changing search ranking parameters to ensure that certain documents or categories of documents appear higher in the result list.

Categorisation
The placing of boundaries around objects that share similarities (e.g. taxonomy).

Clustering
A process employed to generate groupings of related documents by identifying patterns in a document index.

Cognitive search
A description loosely applied by search vendors to applications using machine learning and AI techniques to determine the work context of the user and deliver personalised results.

Collection
A group of objects methodically sorted and placed into a category.

Computational linguistics
The use of computer-based statistical analysis of language to determine patterns and rules that aid semantic understanding.

Concept extraction
The process of determining concepts from text using linguistic analysis.

Connector
A software application that enables a search application to index content in another application.

Controlled vocabulary
An organised list of words, phrases, or some other set employed to identify and retrieve documents.

COTS
Commercial off-the-shelf software.

Crawler
A program used to index documents.

Cross-language search
A query in one language is translated into other indexed languages (often using a multi-lingual thesaurus) so that all documents relevant to the concept of the query are returned no matter what language is used for the content.

Description
A brief summary, generated automatically, that is then included as a description of a document in the list of results.

See also Key sentence

Document
A structured sequence of text information, but often used as a generic description of any content item in a search application.

Document processing
The deconstruction of a document into a form that can be tokenised and indexed.

Document repository
A site where source documents or other content objects are stored, generally a folder or folders.

See also Information source

Early binding
The addition of current access control information to a search index, for later use to control which search results a user is allowed to view.

See also Late binding

Entity extraction
The automatic detection of defined items in a document, such as dates, times, locations, names and acronyms.

Exact match
Two or more words considered mutually inclusive in a search, often by enclosing them in quotation marks—for example, "United Nations".

Facet
Presentation of topic categories on the search user interface to support the refinement of a search query.

Fallout
A quantity representing the percentage of irrelevant hits retrieved in a search.

Federated search
A search carried out across multiple repositories and/or applications.

Field query
A search that is limited to a specific field in a document (e.g. a title or date).

Filter
A function that sets specific criteria for search results.

Freshness
The time period between a document being crawled and the index being updated so that a user will be able to find the document.

Fuzzy search
A search allowing a degree of flexibility for generating hits (i.e., matches that are phonetically or typographically similar).

GitHub
A hosted service widely used to collaborate on software application development and to act as a distribution service.

Golden set
A set of documents used to benchmark search performance that is representative of content that will be searched on a regular basis.

Guided search
A search in which the system prompts the user for information that will refine the search results.

Hit
A search result matching given criteria; sometimes used to denote the number of occurrences of a search term in a document.

Index
List containing data and/or metadata indicating the identity and location of a given file or document.

Index file
A file that stores data in a format capable of retrieval by a search engine.

Ingestion rate
The rate at which documents can be indexed, usually specified in Gb/sec.

Inverse document frequency (IDF)
A measure of the rarity of a given term in a file or document collection.

Inverted file
A list of the words contained within a set of documents, and which document each word is present in.

Inverted index
An index whose entries identify a given word and the documents in which it appears.

Iterative calculation
A calculation utilising a recursive and self-referential algorithm.

Key sentence
A brief statement that effectively summarises a document, often employed to annotate search results.

Keyword
A word used in a query to search for documents.

Keyword search
A search that compares an input word against an index and returns matching results.

Language detection
The indexing process identifies the language (or languages) of the content and assigns it to appropriate language specific indexes.

Late binding
Access permission checking carried out immediately before the presentation of search results to the user.

See also Early binding

Lemmatisation
A process that identifies the root form of words contained within a given document based on grammatical analysis (e.g. run from running).

See also Stemming

Lexical analysis
An analysis that reduces text to a set of discrete words, sentences, and paragraphs.

Linguistics
The study of the structure, use, and development of language.

Linguistic indexing
The classification of a set of words into grammatical classes, such as nouns or verbs.

Meta tag
An HTML command located within the header of a website that displays additional or referential data not present on the page itself.

Metadata
Data that provides information about other data (i.e. is data about data).

Morphologic analysis
The analysis of the structure of language.

Natural language processing
A process that identifies content by attempting to adhere to the rules of a given language.

Natural language query
A search input entered using conventional language (e.g. a sentence).

Parametric search
A search that adheres to predefined attributes present within a given data source.

Parsing
The process of analysing text to determine its semantic structure.

Pattern matching
A type of matching that recognises naturally occurring patterns (word usage, frequency of use, etc.) within a document.

Phrase extraction
The procurement of linguistic concepts, generally phrases, from a given document.

Precision
The quantification of the number of relevant documents returned in a given search.

Proximity searching
A search whose results are returned based on the proximity of given words (e.g. 'pressure' within four words of 'testing').

Query by example
A search in which a previously returned result is used to obtain similar results.

Query transformation
The process of analysing the semantic structure of a query prior to processing in order to improve search performance.

Ranking
A value assigned to a specific result returned for a query—the first item listed has a ranking of 1, the second has a ranking of 2, and so on.

Recall
A percentage representing the relationship between correct results generated by a query and the total number of correct results within an index.

Relevance
The value that a user places on a specific document or item of information.

Search results
The documents or data that are returned from a search.

Search terms
The terms used within a search field.

Semantic analysis
An analysis based upon grammatical or syntactical constraints that attempts to decipher information contained in a document.

Sentiment analysis
The use of natural language processing, computational linguistics, and text analytics to identify and extract subjective information in documents.

Soundex search
A search in which users receive results that are phonetically similar to their query.

Spider
An automated process that provides documents to a data extraction or parsing engine.

See also Crawler

Stemming
A process based on a set of heuristic rules that identifies the root form of words contained thin a given document (e.g. run from running).

See also Lemmatisation

Stop words
Words that are deemed to have no value in an index.

See also Word exclusion

Structured data
Data that can be represented according to specific descriptive parameters—for example, rows and columns in a relational database, or hierarchical nodes in an XML document or fragment.

Summarisation
An automated process for producing a short summary of a document and presenting it in the list of results.

Synonym expansion
Automatically expanding a search by adding synonyms of the query terms derived from a thesaurus.

Syntactic analysis
An analysis capable of associating a word with its respective part of speech by determining its context in a given statement.

Taxonomy
In respect to search, the broad categorisation of objects (typically a tree structure of classifications for a given set of objects) in order to make them easier to retrieve and possibly sort.

Term frequency
A quantity representing how often a term appears in a document.

TF.IDF
The term frequency.inverse document frequency formulation gives a score that is proportional to the number of times a word appears in the document offset by the frequency of the word in the collection of documents.

Thesaurus
A collection of words in a cross-reference system that refers to multiple taxonomies and provides a kind of meta-classification, thereby facilitating document retrieval.

Tokenising
The process of identifying the elements of a sentence, such as phrases, words, abbreviations, and symbols, prior to the creation of an index.

Truncation
Removal of a prefix or suffix.

Unstructured information
Information that is without document or data structure (i.e., cannot be effectively decomposed into constituent elements or chunks for atomic storage and management).

Vector space
A model that enables documents to be ranked for relevance against a query by comparing an algebraic expression of a set of documents with that of the query.

Weight
A value applied to a given area of a search system (e.g. term weighting, which represents its importance with respect to other factors).

Wildcard
A notation, generally an asterisk or question mark, that when used in a query, represents all possible characters (e.g. a search for boo* would return book, boom, boot, etc.).

Word exclusion
A list containing words that will not be indexed—this usually is comprised of words that are excessively common (e.g. a, an, the, etc.).