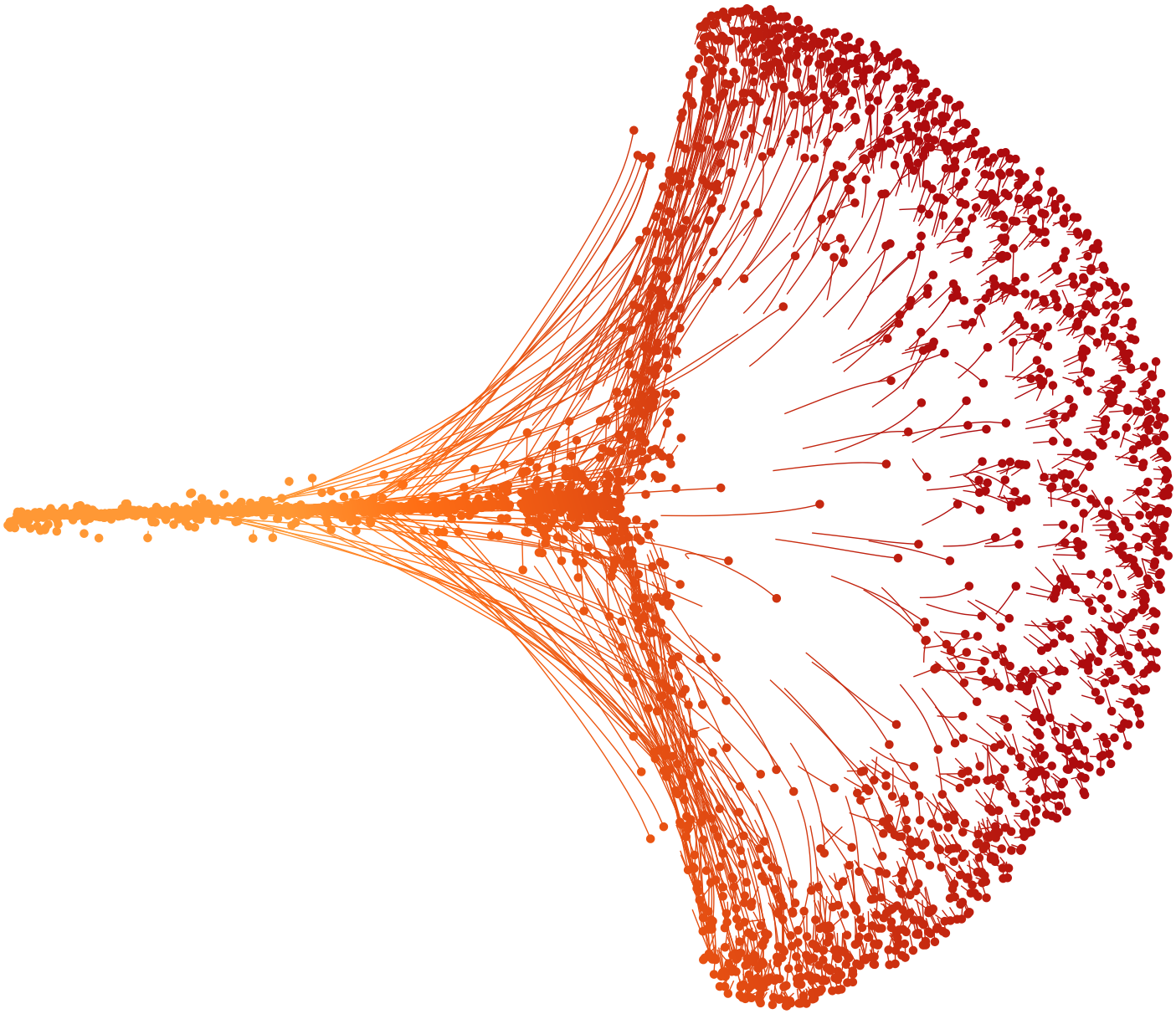


Enterprise Search in 2025

Lucidworks

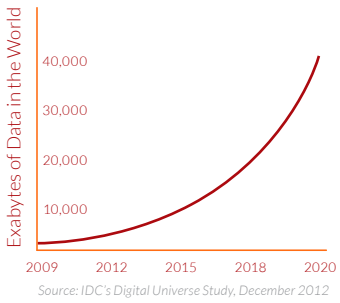
Back in 2011, tech entrepreneur Marc Andreesen declared, that software was eating the world. That technology was quickly taking over entire sectors and industries and transforming them to be more efficient and networked. Four years later, Dries Buytaert got a little more specific, saying "data is eating the world." But even with all the data in the world, you can't do much with it without one crucial component: search. And as data has changed and evolved, so has search.

"Tomorrow's applications will consume multiple sources of data to create a fine-grained context. They will leverage calendar data, location data, historic clickstream data, social contacts, information from wearables, and much more. All that rich data will be used as the input for predictive analytics and personalization services. Eventually, data-driven experiences will be the norm."

— Dries Buytaert, Founder of Drupal

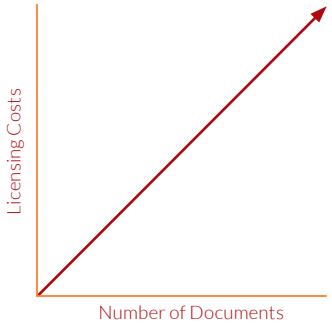
The Rise and Fall of Client-Server Search

The start of modern search, just like the start of modern data, begins with the trusty client-server model. We then started to see breakthrough technologies from companies like Verity, FAST and Endeca. These advances made enterprise search apps possible for many organizations and were later embedded in other enterprise products at the time. Fundamentally, these client-server solutions scaled to meet the challenges of the time: simple file servers, a web server or three, and select but relatively well-structured data.



Source: IDC's Digital Universe Study, December 2012

However, the volume of data exploded and these technologies started to fail and fail big. Not just at a technical level, but on a more fundamental strategic level. Vendors were charging per document indexed so licensing became really expensive as your corpus of data got bigger. Client-server couldn't keep up with indexing data across an entire organization, to do that would require a distributed architecture that scales. Oh and Mr. IT person, "I'm sorry no you can't do this as a nightly job, the data needs to be indexed in real-time" says any boss I've known for the last 20 years or so.



The Internet and companies like Amazon and Google have changed people's expectations of what search should be. As these expectations grew, the need to index more data and provide more context grew. Fast forward to today, where users frequently tap the microphone icon on their phone and say, "Give me a list of four star restaurants near me that are open today" and expect—and get—a real answer.

Yet, ask most business users how they feel about their intranet search (especially SharePoint) and they'll answer that if they lose their magic list of bookmarks they're hosed. When your job depends on using a less than stellar search tool you find workarounds, but somewhere there is a competitor lurking about to data-enable those employees. Bad internal search is inefficiency—a waste of money. Bad external search is lost customers.

Ask most consumers why they use Amazon to buy everything they don't buy retail. The free shipping with Amazon Prime is probably high on the list (along with low prices or maybe price or warranty), fundamentally consumers assume that now matter what they are looking, for they'll find it on Amazon and they'll have it in stock.

To compete, enterprise search and online retail teams alike realize they must have better search that handles their growing data in real-time and provide a personalized experience with **Google and Amazon-like relevance**.

Enter open source, the Apache Lucene project, and Solr. When Doug Cutting started the Lucene project it changed the world. But not overnight. In Lucene, you had an index that could be both easily embedded and easily replicated across a network. Later, when Apache Solr extended Lucene into a full-fledged search solution, it changed the world of search forever.

Solr Killed the Client-Server Stars

The debut of Solr meant that anyone anywhere could have a scalable search solution—as long as they were willing to setup hardware and find a way to manage it. Solr gave anyone a way to provide high-end relevant search—as long as they took on the ingestion and UI development themselves.

Solr destroyed the pre-existing enterprise search marketplace. All of the client-server search vendors got out fast. Its simple in hindsight: the delta between shoehorning these client-server solutions into something that handles today's requirements at today's scale combined with competing against an open source solution would have required too much investment. The smart money was to sell out to a bigger vendor quickly while you had a long tail of customers and an entrenched install base.

Microsoft acquired FAST. HP got Autonomy (which had purchased Verity). Oracle bought Endeca. Did the acquiring firms put a lot of investment into these technologies to bring them to the next level or did they simply raise the price and milk the long tail? We'll let you guess. While this big squeeze started, other vendors like Attivio found new focuses for their product and basically abandoned their original enterprise search pedigree.

Meanwhile Google realized they didn't like being an enterprise hardware vendor let alone providing the service levels needed (and we're not even mentioning their outdated pricing model). Google announced the end-of-life for their Google Search Appliance product leaving organizations all over the world in a lurch.

Other solutions developed, but the Solr ecosystem became the unmatched winner of the search market. Search 1.0 was over and Solr won.

Nov 2005
Autonomy buys Verity

Jan 2008
Microsoft buys Verity FAST

Oct 2011
HP buys Autonomy

Oct 2011
Oracle buys Endeca

Feb 2016
Google EOLs GSA

New Data

It used to be that email was the only example given when we talked about unstructured data. When big data was first branded the most common objection companies had was "We don't have big data." This was true so long as they only looked at their well-structured data sources and didn't try to analyze their data in aggregate.

Solving search was the first big data application. Each individual webserver could have said, "We don't have big data, we only have a few HTML pages and a handful of images." However, when Sergey Brin and Larry Page told their Stanford professor that they wanted to download and analyze the entire web in 1996, this was already a big storage and scaling problem. They needed new distributed technologies.

As it turns out, nearly all substantial organizations have email, log data, sensor data, network data, intrusion data, A/V data, phone data, CRM data, CMS Data, social media data, and all kinds of semi-structured and unstructured data that was next to impossible to analyze.

The initial problem with this data was that there was no way to adequately store it. But storage got cheaper and distributed filesystems came along and fixed that. Then the problem was that there was no way to capture it quickly enough without affecting the system that produced it or the overall network. Networks got faster, CPUs got faster and grew more cores and event-streaming software made real-time capture efficient. With faster and cheaper storage conquered, networking advances, distributed filesystems, and now streaming technologies, there was no problem with either a storage or capture.

With these new types of data, new techniques were needed to process and analyze it. Many of the techniques necessary had existed for decades in the field of artificial intelligence, advanced mathematics, and statistics. These techniques would become known as machine learning.

There were companies using machine learning and statistics before the current era of big data. However just like distributed filesystems and streaming technology, these technologies had been held back due to cost-prohibitive proprietary licensing and hardware costs. Cloud technologies would address the latter through pay-as-you-go and burst capacity. With open source technologies like Apache Spark and MLlib, distributed computing, artificial intelligence, and machine learning are available to any company with the right expertise.

There are well established use cases for machine learning in retail and finance from recommendation systems/offer management, fraud/intrusion detection and risk management/price optimization. However, we're still in the early days and companies are finding new ways to use their unstructured, semi-structured, and structured data every day. Additionally, some traditionally batch cases are becoming real-time through these new technologies.

Big/New Data Meets Search

The last generation of Hadoop and Spark deployments were composed of so-called data lakes in which one would "dump the data on the filesystem and figure it out later." As big data use cases developed, the need for an index quickly become apparent. In order to find the proper working set, every iteration of the same analytic searches through the entire data corpus every time—over and over.

With Hadoop deployments, this repeated full-scan parsing of files resulted in a big performance hit. "Hadoop is slow" is the frequent complaint. With Apache Spark, these processes needed a lot more nodes to avoid out of memory errors. "Spark is expensive" is the frequent complaint.

By using an index, storage is optimized since only the required data is loaded into memory as part of the working set to be analyzed.

By using an index, storage is optimized since only the required data is loaded into memory as part of the working set to be analyzed. This is a good compromise between the overly-structured RDBMS (i.e. Oracle/SQL Server) and the under-structured Hadoop Distributed Filesystem (HDFS) so that the right data is loaded quickly and efficiently.

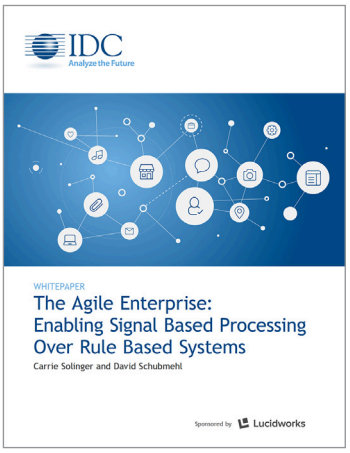
This hybrid approach becomes even more critical as systems move to real-time processing. Lots of data coming in means finding the needle in the haystack is a bigger task. Ultimately this is an issue of using the best tool for the job. Finding the working set is a Solr problem, analyzing the working set is a Spark problem.

Data Gets Personal

Google Now tells a customer the in-traffic driving time to their favorite Vietnamese place that they go to every Saturday. The user never asked for this information, Google Now just figured it out based on their driving patterns, likes, and other data. Amazon shows me Instant Pot recipe books on the front page because I search for Instant Pots or possibly purchased one in the past.

Putting this personalized view together is a result of a combination of signals captured from the user like clickstream and location data as well as rules. The signals are personal data and that the customer clicked on or drove by a particular store location. The rule is that they must have driven by it or clicked on it more than 3 times to cause the search result to boost or be displayed on the front page for a given user.

Customers are profiled, queries are answered with context, demographic data is no longer king. It isn't enough to band someone in "males living in the southeast US in the 50k-100k income bracket" and show you sports-related content. Instead, companies with the edge are targeting a customer specifically. This requires a lot of data and sophisticated software that can handle both aggregating signals and collating rules and applying them to search results.



Want to learn more about rules verses signal based relevancy?
Download the IDC whitepaper »

The Next Wave of Data and Search

Based on this history and these trends what is coming next?

Cloud Hybrid

Data is moving to the cloud. There is still a lot of data on-premise and search will be one of the last things to go 100% cloud—and will be one of the only real hybrid cloud technologies. The reason is simple, businesses need to index the data that isn't in the cloud. Businesses also need to index data that is in the cloud. It makes sense to have search technologies that work behind the corporate firewall as well as in a cloud or Virtual Private Cloud (VPC).

Talking to the Machine (and the Machine Talks Back)

Conversational search and voice search are going to take over search entirely. Whether you're typing "What are the 2017 Q1 sales figures" or saying "Where is my car?" nearly all search will be conversational in the next few years. According to Google, we're already at around 15%. I may still type a movie title into Netflix or an exact product name into Amazon but when I don't know exactly what I want I'll just ask "Show me newly released action flicks from 2017 that are rated at least 3 stars and have a decent plot" or "I need something to rehydrate beans in less than a few hours."

In addition to speech to text features, this capability will require signal processing, natural language processing and machine learning at a level not typically deployed.

Predictive Search

The best thing about Google Now is that I don't even have to ask. The current Amazon front page is a combination of one's history, promotions, and recommendations based on what similar customers purchased. Relative to what current technology is capable of, it is still a pretty broad hammer approach to search, promotion and personalization.

Signals like one's interest, past purchases, and characteristics are becoming much more personal and predictive. Rather than a simple grouping "similar customer" a more predictive model like "those interested in this and that who purchased X and Y are 30% likely to buy Z if shown this promotion" and automatic A/B testing and refinement will automate away the need for most of the promotions and demographic recommendations.

Signals like one's interest, past purchases, and characteristics are becoming much more personal and predictive.

By the time a customer has any history at all, the machine will automatically make recommendations up front. The best customers may never need to use the search box again!

Ubiquitous Search

With tools like Alexa (Amazon Echo) and Internet of Things (IoT) devices deployed throughout the house, expect to see more search that no one realizes is search. Sure I have various devices like dishwashers and refrigerators, but am I using them? How much am I using them? If I don't open the refrigerator much then maybe I eat out a lot. If I use the oven often, then maybe I like baking. Is my food going bad, how can I optimize my purchases to prevent that? What Fitbit and My Fitness Pal are doing for personal fitness, IoT, machine learning, and ubiquitous search will do for the rest of life itself.

Obstacles to the Future

It isn't all roses for everyone. There will be a lot of dead companies along the way. Failure will be littered with:

1. **Bad data** - Garbage in, garbage out. Machine learning will not help you if you don't sort things correctly.
2. **Bad science** - Machine learning isn't magic. If you produce random data and feed it into a neural network it will find relationships between the datapoints. It will take good expertise to form the questions that your software tries to answer with good data.

3. **Creepy companies** - While use cases are many, there is a fine line between being helpful and being creepy or obnoxious. Google tends to roll out new personalized features quietly and initially unobtrusively. Companies that fail to heed this may find themselves facing customer backlash or even lawsuits.
4. **Obsolescence** - Technology moves fast and the data that feeds it is ever increasing in volume. If you're stuck on Siebel and Webtrends, you're probably not going to make personalized conversational search happen. Your competitor will find it easy to disrupt you. Good IT balances the risk of new technology without obstructing progress.

Driving Forward

What can you do to start making the future happen today?

Deploy software that is cloud-ready but can deploy on-premise. Most businesses have a ton of asset data that is behind the corporate firewall. It may be cost or performance prohibitive to index this in the cloud. There may be other considerations as well. At the same time your search software shouldn't be a limiting factor preventing you from deploying cloud capabilities. The ideal software can go on-premise or in the cloud. You shouldn't have to chose.

Avoid pitfalls. There are a lot of **ways you can paint yourself into a corner**. Understand how search projects go wrong (i.e. bad data, bad schema, poor relevance, poor resource planning, rolling your own). Don't do that.

Hire the right expertise. Becoming a data scientist is as easy as putting "Data Scientist" on your resume. Actually understanding statistics, machine learning, and how to use NLP are another matter entirely. You need the right mix of expertise from the business, development, and mathematical backgrounds to drive the next era of search.

You need the right mix of expertise from the business, development and mathematical backgrounds to drive the next era of search.

Create a permanent technology refresh plan. Technology doesn't stand still. Customer expectations don't stand still. A few years ago, people were content to navigate an online retail site to exactly the right category then find the brand then find the item. These days, if it isn't the top result it might as well not exist.

Deploy new capabilities. If you're not profiling how customers use your site, start. If you're not profiling how customer use search, start. If you haven't married search with purchase history, start. If your data is too hard to get to, fix that. Moreover, keep up with **current trends** and avoid falling behind the curve.

Use smart A/B testing. For big retailers or even companies like Salesforce, a small change in relevancy can have a big effect. Even the best QA can't predict what

might happen when real customers are faced with a change in ranking algorithms. Salesforce tests this side-by-side. You should find a way that balances risk without hindering progress including testing on real customer searches and doing a rollout that controls risk.

Work on data quality. If your data isn't good, then finding it doesn't do anyone any good. Data quality is job #1 for any modern business. Whether it is search or machine learning, you need a good approach to making sure the input is good!

Bottom Line

Data is eating the world and search is the key to finding the data you need. The enterprise search industry is consolidating and moving to technologies built around Lucene and Solr. In the next few years we'll see nearly all search become voice, conversational, and predictive. Search will surround everything we do and the right combination of signal capture, machine learning, and rules are essential to making that work. Fortunately, much of the technology to drive this is available to us today!

Need a hand?

Lucidworks has been there and done that. We can help you navigate these trends as well as architect and deploy a solution that is scalable, relevant, and future-proof. If you find yourself looking at all this and wondering where to start, contact us at **lucidworks.com/contact** or give us a call at **415-329-6515**.