

How Full-Text Search Engines Improve Productivity At Work

Article contributed
by [dtSearch®](#)


Entering the era of private space travel, you'd think that even if the rest of us can't yet launch into orbit, at the very least we could make our lives easier through instant search across terabytes of up-to-date enterprise data from any place we may happen to visit on this planet. The good news is, with enterprise search software and a search index, we can make Earth that much more habitable while we await our own spacecraft.

Now you may look up at the stars and wonder: what is a search index? A search index is not like the type of index you'd find at the end of a large book. Instead, it is simply an internal tool for storing each unique word and each unique number across an enterprise dataset, and the locations of all words and numbers in the data. The sole purpose of the search index is for the search engine to comprehensively search across everything, processing as many search requests as come in at any one time.

For this specific example, each search index can hold up to one terabyte, and there are no limits on the number of terabyte-size indexes the search engine can build and simultaneously cover for concurrent search requests. Supported data types include Microsoft Office files, PDFs, compressed archives like ZIP or RAR, emails plus attachments, databases, web-based formats and more. And getting all of this into the index couldn't be easier.

To index, just point to the folders, emails, etc. you want to cover, and the search engine will do the rest. No need to even tell the software what types of data it is indexing. In fact, the search engine can figure out for itself if an item is a PowerPoint versus a OneNote file versus an email by looking inside each binary file (in this article I'm using specific examples from dtSearch although concepts like a search index have general applicability.)

Because the search engine looks inside each binary file to determine the file type, it doesn't even matter if a file has the wrong file extension. You can have an Access database saved with an Excel file extension, and a PDF saved with a Word extension, and the search engine will sort all that out. If you have files nested inside other files, the search engine will figure that out too. An email with a ZIP attachment containing a Word document with a fully embedded Excel spreadsheet is no problem.



Because this is a big planet, these search options work not only for English, but also for hundreds of international languages via Unicode.

After indexing, the search engine can run from a secure online environment such as a Windows IIS web server. The server can be "on premises" or in the cloud such as on Azure or AWS, enabling search from any device with a site connection and a web browser. For enterprise search, the site will require proper security credentials. Once users are in, search requests can proceed in a stateless manner, supporting an unlimited number of concurrent instant search threads.

Beyond individual word search, the search engine has over 25 different types of full-text or metadata-specific features, such as Boolean (and/or/not), phrase, proximity (before or after), directed proximity (before only), concept/synonym, date or date range, number or numeric range, wildcard, or any combination of these. Fuzzy searching adjustable from 1 to 10 can sift through typographical errors as often occur in email text or in OCR'ed PDFs. The search engine can even identify any credit card numbers in text.

Because this is a big planet, these search options work not only for English, but also for hundreds of international languages via Unicode. After a search request, the search engine can relevancy-rank retrieved data by hit term density and rarity. That means that if you search for planets, asteroids or comets, and planets and asteroids are all over the data, but comets are much rarer, then comets will get a higher relevancy score. Denser mentions of comets inside a single document or email will rank even more highly.

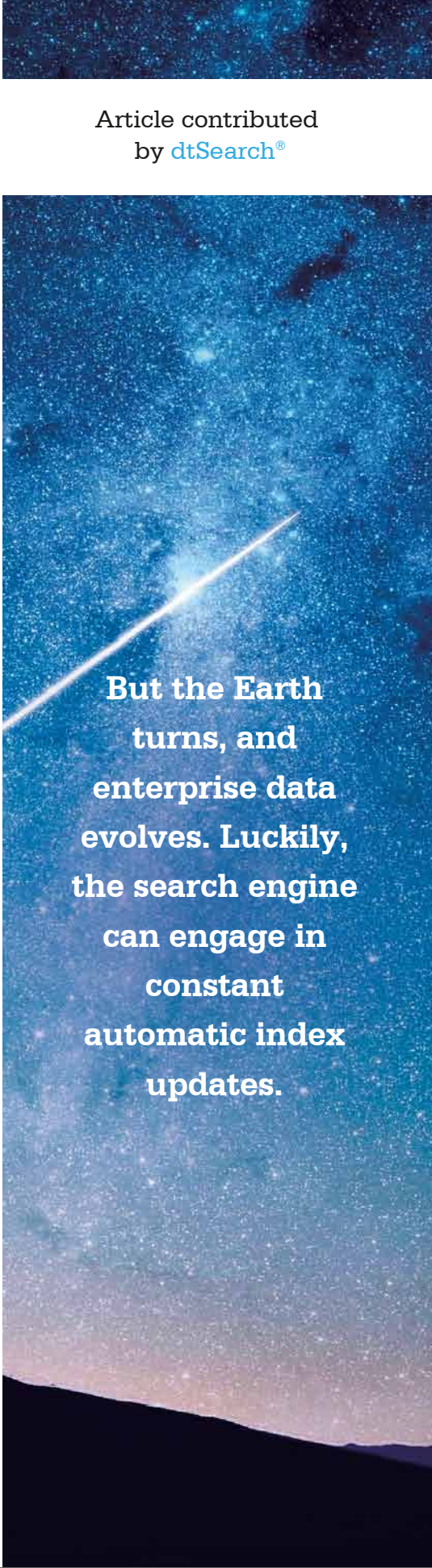
The search engine has multiple sorting options, including not only by relevancy but also by filename, file location, and other metadata information. Each sorting option is like a different window into the data, and you can instantly re-sort with a single click. After a search request, the search engine can display the full text of retrieved files, emails and the like with highlighted hits.

But the Earth turns, and enterprise data evolves. Luckily, the search engine can engage in constant automatic index updates. Rather than re-indexing everything from scratch, the index updates can account for only new files, deleted files, or files with new edits. Importantly, automatic updates can proceed *without* affecting concurrent searching.

A caching option further lets the indexer save a full copy of original files, emails, etc. as part of the index itself. With caching, the search engine can still display retrieved files, even if the originals are unavailable. That way, even if some data goes offline, searching with highlighted hits can continue uninterrupted.

Life is a little smoother for the rest of us stuck here on earth.

Article contributed
by [dtSearch®](#)



**But the Earth
turns, and
enterprise data
evolves. Luckily,
the search engine
can engage in
constant
automatic index
updates.**