Rev Up Hybrid Work Productivity with Search Engine Deployment and Caching

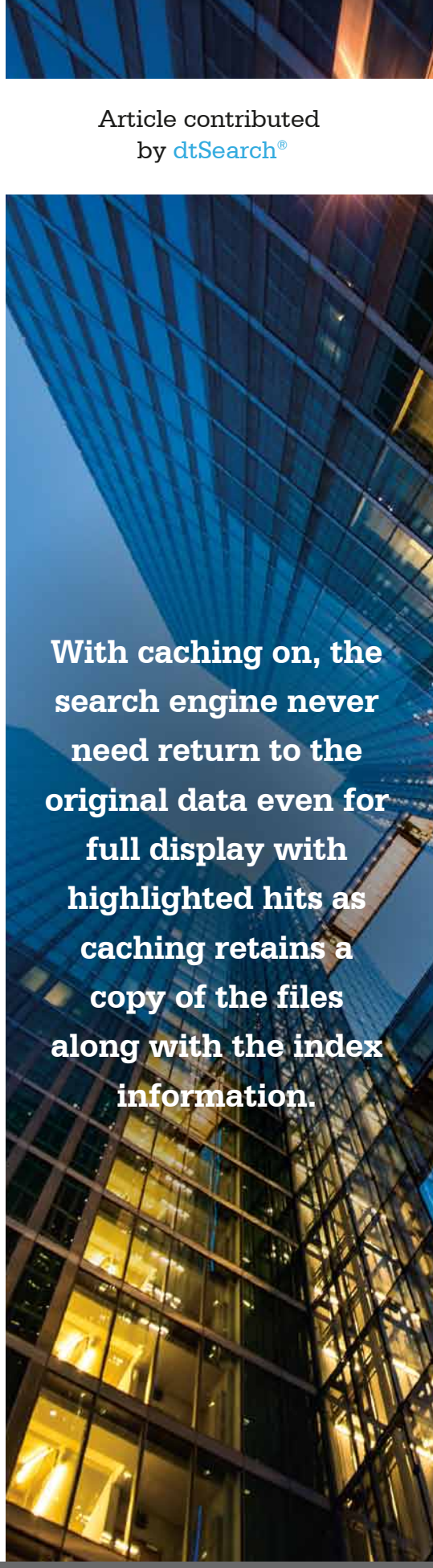
Article contributed by dtSearch®

We all like working from home sometimes. But what happens when you're on a home day and you suddenly realize you need a file from the office? An enterprise search engine can bridge that gap.

Enterprise search represents a category of applications like dtSearch® as opposed to scan-the-internet search products like Google. Enterprise search instantly searches terabytes of mixed organization data like PDFs, Microsoft Office files, emails, compression formats, web-based data, etc. after first indexing the data. An index is just an internal tool that the search engine builds to efficiently search the data. Indexing is easy. Simply point to the relevant folders and the like to index, and the search engine will do the rest.

To build its index, the search engine will review the binary formats of all files in selected folders and subfolders. PDF, Word, Excel, Access, PowerPoint, OneNote, email, all have very different parsing specifications. The search engine has built-in document filters to correctly identify the binary format of each file and apply the right parsing specification to ascertain the text and metadata. This document filters' processing is nothing if not thorough.

- Multilayered nested binary formats like an email with a ZIP or RAR attachment that includes a Word document with an Excel spreadsheet buried inside are no problem for the document filters.
- Metadata that may take a whole lot of clicking around inside a file's native application to see is immediately available to the document filters from the binary format.
- Black text against a black background or white text against a white background that might be nearly impossible to spot inside a file's native application is just regular text inside the binary format.
- Mismatched file extensions, like a PDF saved with a .DOCX extension, will not trip up the document filters since these can look inside the binary format to determine the correct file type.
- The document filters can even spot PDFs that look like "normal" PDFs sitting there in the file system but are really "image only" and hence require OCR by an application like Adobe Acrobat for text processing.



Once indexing is complete, the search engine can instantly search terabytes of mixed data including over 25 different search features. After a search runs, the search engine can display the full text of retrieved items with highlighted hits. Within these general parameters, however, there are very different options for deploying the search engine in a hybrid environment.

(1) The search engine can execute from a secure web server, providing convenient data access to both in-office and out-of-office workers. The web server itself can run from a physical server inside the organization or from a cloud hosting platform like Azure or AWS. Web-based search can operate in a completely stateless manner, with no limit on the number of concurrent search threads. Instant concurrent search can continue uninterrupted while indexes automatically update to reflect new content.

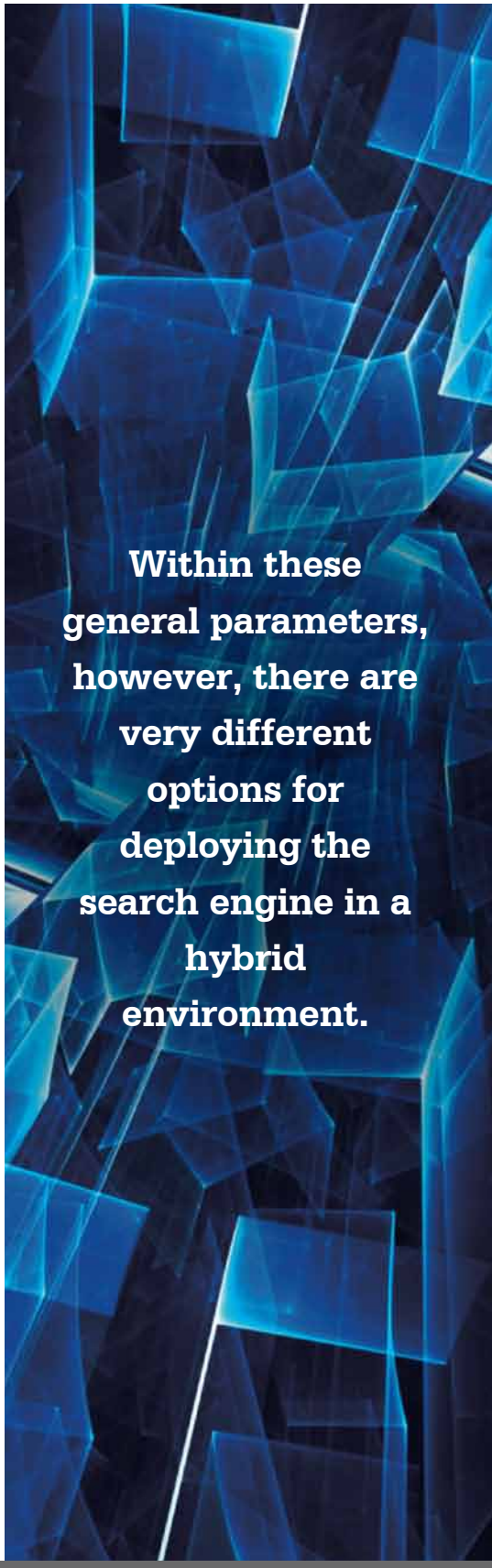
(2) The search engine can run in a classic Windows network-type environment, including remote access for those offsite. In addition to instant concurrent indexed searching, this configuration can also include classic Windows features, like enabling an end-user to launch a Word document in Microsoft Word after retrieving it in a search.

(3) The search engine can run in one of the above configurations in-office while operating separately from a laptop outside the office. The remote laptop can contain copies of the files themselves, or just include the indexes alone if caching is on.

What is caching? Let's take a search for *hybrid workforce* within 13 words of *worker productivity* or *efficiency* in a file that also mentions *coffee break* and not *candy corn*. The search engine can instantly find files that match this search request. But to go one step further and display a full copy of retrieved files with highlighted hits, the search engine has to return to the original data.

With caching on, the search engine never need return to the original data even for full display with highlighted hits as caching retains a copy of the files along with the index information. The caching setting works well in all above hybrid search configurations. In the first two configurations, caching can make full file display snappier, particularly if the alternative is waiting on a server to pull up a remote 675-page PDF. In the third option, caching allows the remote worker to just copy the relevant indexes without worrying about also copying the underlying files.

Article contributed by dtSearch®



Article contributed by dtSearch®

Any of the above configuration options will rev up in-office and remote work productivity. Some quick search tips follow to take this even farther.

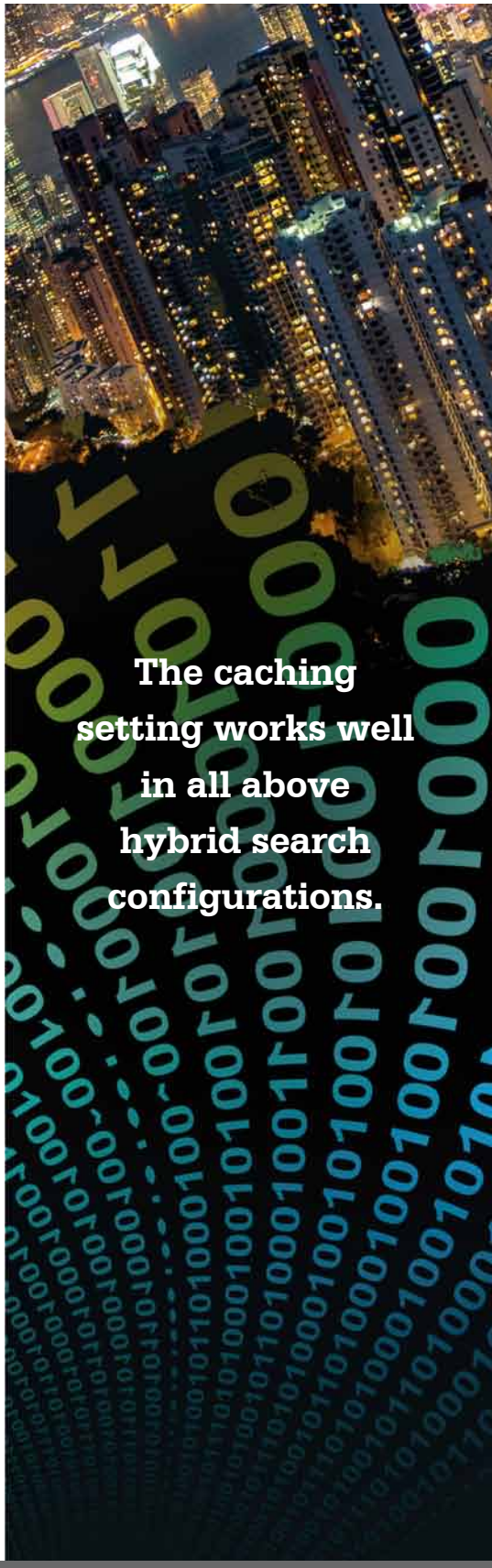
Tip #1: For both structured and unstructured natural language search requests, the search engine will relevancy rank retrieved data via a "vector space" algorithm that takes into account search term rarity and density. If *efficiency* is prevalent across indexed data but *coffee* pops up in just a few places, *coffee* files (especially files with denser *coffee* mentions) would get a higher relevancy ranking, putting these files first. But you can further adjust relevancy to add your own custom positive or negative weighting beyond the default ranking, like giving *hybrid* an extra positive ranking if it appears at the top or bottom of a file or an extra negative weight if it appears in certain metadata. These techniques are particularly useful when a search returns a huge number of files.

Tip #2: You can instantly re-sort search results by a non-relevancy criterion like file date or file location to get a different window into search results. This tip is also helpful when a search retrieves a large quantity of files.

Tip #3: Activate fuzzy searching (adjustable from 0 to 10) to sift through minor typographical errors such as *hybnid* in a search for *hybrid*. Such typos can easily pop up in OCR'ed text or in emails, and a low level of fuzzy searching can sift right through them.

Tip #4: Add in concept searching to a search request not only to locate general synonyms like *auto* for *car* but also to find custom synonyms relevant to your own work. If you are working on *ProjectCDE*, and the same project had the previous names *ProjectABC* and *ProjectBCD*, all of these can be synonyms for purposes of concept search.

Tip #5: In addition to word-oriented searching, the search engine supports numeric-oriented searching. Add in a search for a specific part number or number range to a search request. Or add in a specific date or date range search, even extending automatically to different date formats. For example, a date range search covering *March 5, 2022* to *April 20, 2023* would retrieve a reference to *11/16/22* in the full text or metadata. The search engine can also flag credit card numbers in indexed data, so that a network administrator can identify files that might accidentally include a credit card in "open" data.



The caching setting works well in all above hybrid search configurations.