Unlock the Value of Data

Article contributed by dtSearch®

If you look at raw data in binary format, you'll typically see just a jumble of binary codes. This applies to raw output from almost any format: Outlook, Exchange, PDF, Microsoft Word, Excel, Access, PowerPoint, OneNote, etc. In fact, it can be hard to make out words and sentences at all in binary data. The first step to unlocking the value of data is to parse these formats to get to the underlying main body text and metadata.

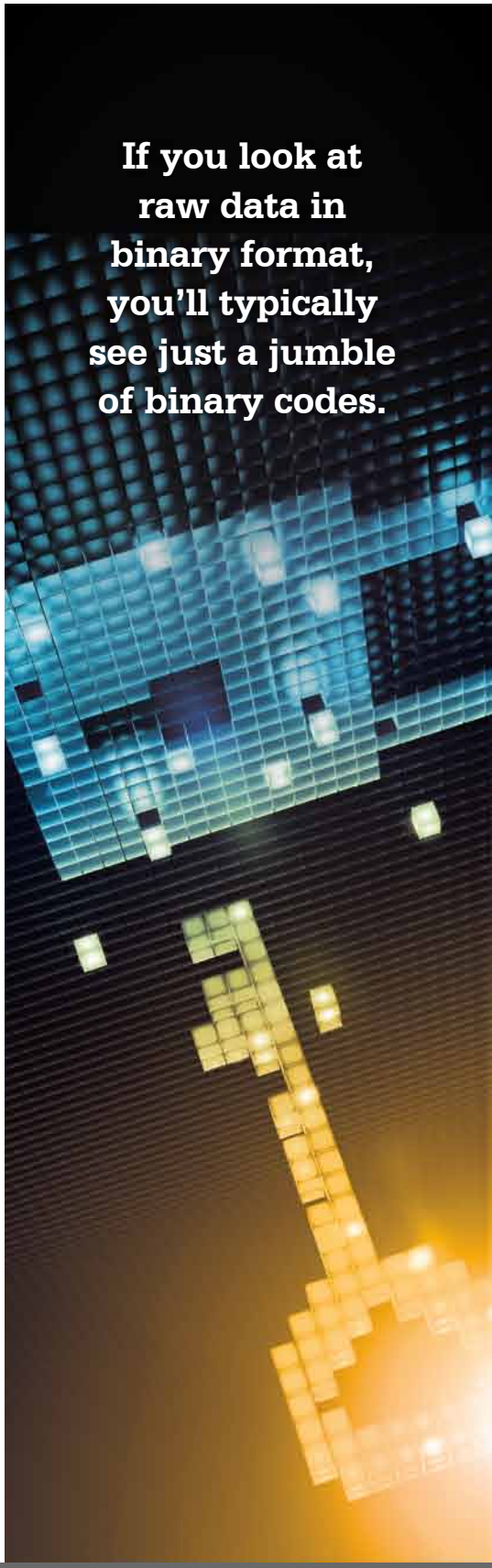
You could pull up each file in its native application to review the main text and metadata – viewing emails in Outlook or Exchange, word processing documents in Word, etc. That works if you are just reviewing a handful of files. But if you have millions of files to review, retrieving each in its native application is not feasible.

An alternative to pulling up each file in its native application is to directly parse the binary formats. Document filters are the component of a search engine like dtSearch® that does that. (Please visit dtSearch.com to download a fully-functional 30-day evaluation copy.) In fact, dtSearch's own proprietary document filters are built in across the entire dtSearch product line.

The first step for the document filters is to figure out the file type of each item. The parsing requirements for a Microsoft Word format differ greatly from the parsing requirements for a PDF, not to mention email formats and the like. You might think that the document filters could use a file extension like .PDF or .DOCX to figure out the file type. However, file extensions are hardly a bulletproof indicator of file type. It is all too easy, for example, to save a PDF with a OneNote extension, and an Access file with an email file extension.

So how do the document filters determine file type? By looking inside each binary file itself. For compressed archives like ZIP or RAR, the document filters will decompress the data and then look inside each individual binary file to determine the file type. Sometimes a binary format can consist of more than one file type, like an email with a ZIP attachment including a PDF and a Word document, and an Excel spreadsheet nested inside the Word document. The document filters need to figure out the relevant file type and parsing specification at each level.

After the document filters parse the underlying data, the next step to unlocking the value of the data is to have the search engine build an index across everything. (dtSearch can actually search without an index, but that process is slower.) After dtSearch builds an index, search time is typically instant, even across terabytes. Indexing enables instant search by effectively pre-processing each unique word and number in the data and that word or number's location in the data.



Indexing is not hard to do. Actually, the process is automatic. All you need to do is point to the various drives and folders to index, and the search engine does everything else. After indexing, even complex search requests like subject field includes *ProjectX* and full-text contains *gold bars and treasure trove and (platinum within 70 words of silver) and not pirate booty* can proceed instantly.

And dtSearch has over 25 different search options. For example, you could use a built-in thesaurus or user-defined synonyms to find similar concepts to those in the search request. Fuzzy searching, adjustable from 1 to 10, will sift through potential misspelling. If *platinum* is mistyped *platimum* in an email, or *silver* ends up as *silven* in an OCR'ed PDF, fuzzy searching will still find that. Natural language searching let you enter unformatted search requests. Directed proximity search lets you enter a word or phrase only if appears X words before another word or phrase. Instead of looking for *platinum within 70 words of silver*, you could limit a search to files that contain a *silver* mention within 34 words before a *platinum* mention.

Other general options include numeric range searching, regular expression searching, date and date range searching, and adjustable relevancy-ranking. dtSearch also has advanced forensics-oriented search options like automatic recognition of credit card numbers in text and file hash value generation and search. And dtSearch has Unicode support, so you can search not only English and other European text but also double-byte Asian characters and right-to-left languages like Hebrew and Arabic. dtSearch can even look for certain emojis in text 😊

After a search, the program goes back to each original email, document and the like and displays that with highlighted hits. Beyond individual search, dtSearch can also run in a multiuser environment, either offline or online, including in the Azure or AWS cloud. For multiuser environments, dtSearch will process each search thread independently, with no limits on the number of search threads that can proceed instantly and concurrently.

dtSearch can update indexes as often as you like to accommodate the addition of new data to a data collection. When dtSearch updates an index, it doesn't have to rebuild the index "from scratch." Rather, it can just reindex new or modified files. And updating an index does not affect continuing concurrent searching, enabling continual unlocking of minute-by-minute updated data.

Article contributed by dtSearch®