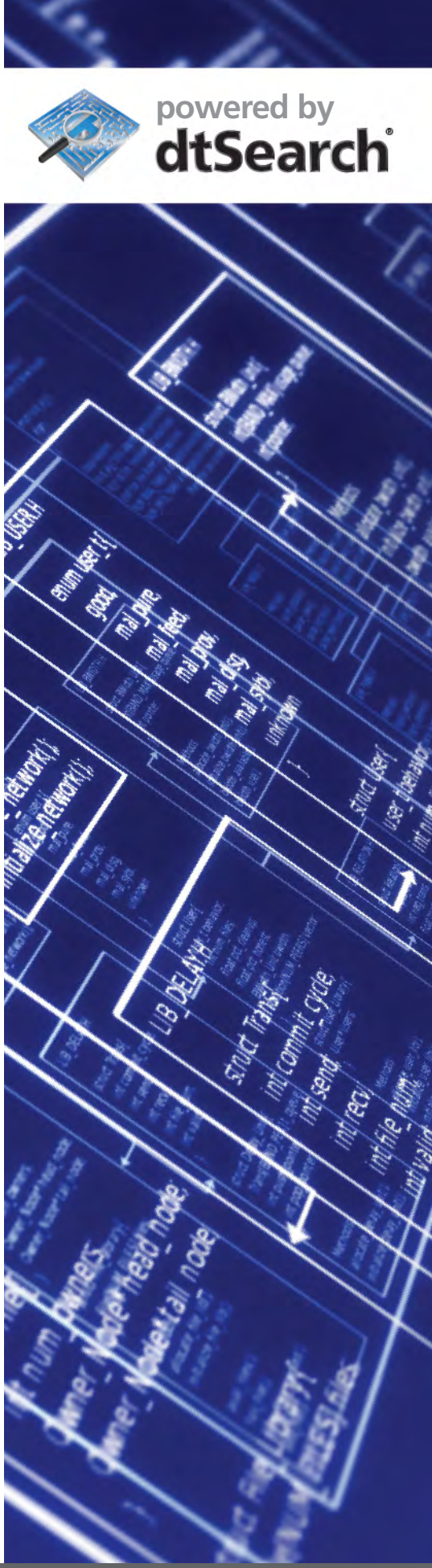
Optimize searching to find every 'smoking gun'



Search engines like dtSearch sift through terabytes of random data: Microsoft "Office" formats; non-Microsoft "Office" formats; PDFs and PDF portfolios; compression formats; web data; other databases; email in multiple formats; and even multilayer nested email attachments.

Fortunately, you don't have to tell dtSearch what types of data it is searching. dtSearch can figure that out for itself. After a search, dtSearch can display retrieved files, emails and other data with highlighted hits.

While able to search terabytes, dtSearch products are not forensics and e-discovery specific applications. For example, dtSearch Desktop with Spider provides general-purpose desktop search. dtSearch Network with Spider offers general-purpose network search. And dtSearch Web with Spider publishes and instantly searches data on an Internet or Intranet server.

Although dtSearch is not itself forensics or e-discovery-specific, many **forensics** and **e-discovery** products do embed dtSearch's core developer module, the dtSearch Engine, both for data support (so-called document filters) and for search functionality.

No matter how powerful the search engine, a flawed search request achieves flawed search results. If you are searching a large volume of unfamiliar data, a flawed search request may fail to retrieve a key "smoking gun." Worse, you may not even realize that a "smoking gun" was present at all. This article seeks to provide search tips to reduce the chances this will happen.

Search Tip #1: In search requests containing two or more Boolean connectors, use parentheses for grouping. The key Boolean connectors are: *and, or, not, w/, pre/*. Most people are familiar with *and, or, not*. *W/* finds a word or phrase within x words of another word or phrase: *preponderance w/15 evidence*. *Pre/* finds a word or phrase within x words before another word of phrase: *preponderance pre/8 evidence*.

When a search request has more than one Boolean connector, use parentheses to clarify. Without such clarification, *Germany w/3 France or Italy* could be either *(Germany w/3 France) or Italy* or *Germany w/3 (France or Italy)*. Likewise, *alphabet or noodle and not soup* has a very different meaning as *alphabet or (noodle and not soup)* than *(alphabet or noodle) and not soup*.

The parentheses rule has an exception for search requests containing a series of terms linked only by *or* connectors, or a series of terms linked only by *and* connectors. (See also **search for list of words**.) But the safer way to proceed is to use parenthesis any time you see two or more connectors.

Search Tip #2: In all searches, use quotation marks around phrases. This tip is particularly important when a phrase includes one of the *and, or, not* Boolean connectors. For example, the phrase *clear and convincing evidence* includes the connector *and*. To search for the whole phrase as a phrase, use quotation marks: *("clear and convincing evidence" and not "preponderance of the evidence") w/55 verdict.*

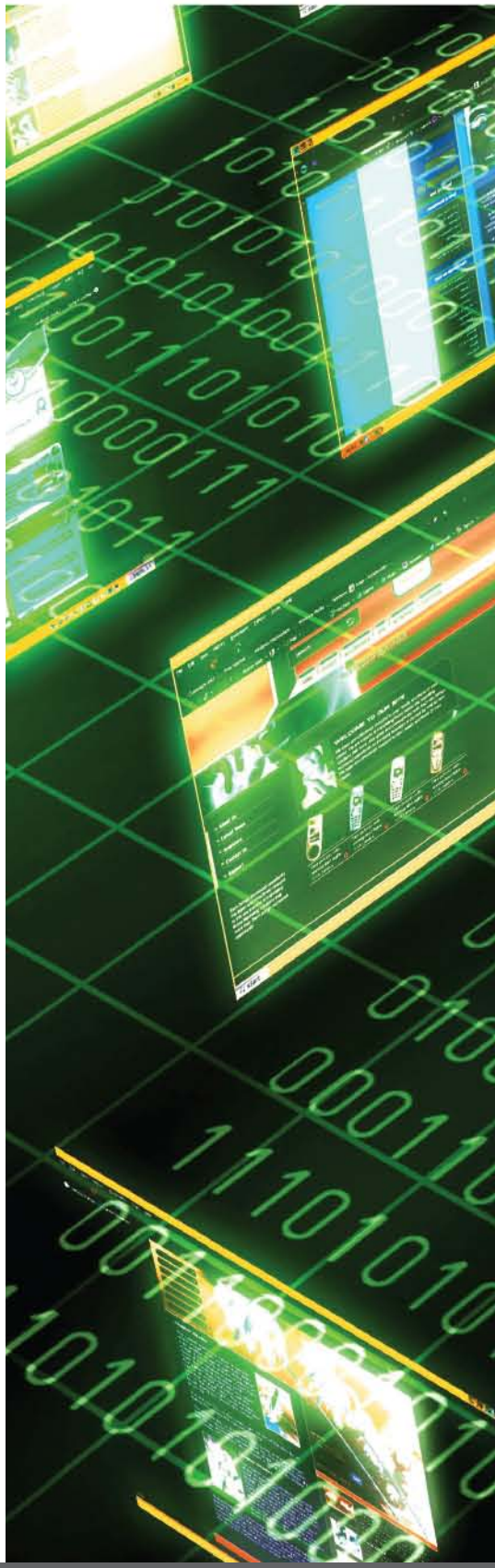
Search Tip #3: Store OCR (optical character recognition) output in "searchable image" PDF format. If you are working with paper documents or images containing text, use a program like Adobe Acrobat to OCR in the "searchable image" (or "image with hidden text") PDF format. This format preserves the complete original scanned image, while adding the OCR'ed text for search engines.

With a "searchable image" PDF, dtSearch can use Adobe Reader to display the full original document or other image, including handwritten notes, drawings and the like. At the same time, dtSearch can, through its Adobe Reader hit-highlighting **plug-in**, highlight hits "beneath" the image of the document. In this way, "searchable image" PDF becomes as close as you can get in the OCR world to having your cake and eating it too. (**More**)

Search Tip #4: Activate fuzzy searching at a low level to sift through potential scanning and typographical errors. This tip is important not only with OCR'ed text, but also with emails, where everyone mistypes. Fuzzy searching looks for deviations in search term letters. With fuzzy searching set to a level of 1, a search for *alphabet* would find *alphaqet*; with a fuzziness of 3, it would find both *alphaqet* and *alpkaqet*. (**More**)

Search Tip #5: Take advantage of the user-defined thesaurus to find synonyms that apply to your case. For example, *Frank* and *Jones* would not be synonyms covered by dtSearch's built-in English language thesaurus. But if you are working on the *Frank Jones* case, you may want to make them synonyms for purposes of your case. And the same principle applies to technical jargon like *airbags* and *SRS*. (**More**)

Search Tip #6: While you may not need advanced search techniques in all cases, at least have a general idea of the range of options. You never know when retrieving all credit cards or email addresses in data, or searching double-byte Chinese text without spacing, or applying positive weighting to a search term in one field and negative weighting in a different field, will make or break a case. **Search types** overview and special **forensics and e-discovery search options** include an overview of advanced search types.



When search engines instantly search terabytes of text, they do so by automatically building an index that stores each unique word and its location (or locations) in the data. In addition to indexed search, some search engines like dtSearch give you the option of unindexed (single pass, no index) searching. Taking one step back from searching, following are some tips for indexing.

Indexing Tip #1: Build an index. Unindexed searching is almost never more efficient. While indexing is much slower than searching, the time it takes to build an index and then search for multiple search terms (as is typical in forensics and e-discovery) is significantly less than the time it takes to run multiple unindexed search terms. And once the index is in place, if you think of more search terms, additional search time is pretty much instantaneous.

Indexing Tip #2: Watch for encrypted files. After building an index, dtSearch's "off the shelf" products, for example, create a log of encrypted files dtSearch cannot read. Take a look at this log so you know what you need to separately decrypt and run again through dtSearch. (**More**)

Indexing Tip #3: Access emails directly as PST, OST, MSG etc. files, instead of going through Outlook/MAPI. If you are not searching your own personal email collection (and sometimes even if you are searching your own emails and have a large collection), it is much more efficient to bypass the Outlook/MAPI "middleman," and directly access the data. (**More**) And don't forget fuzzy searching to sift through potential typographical errors in emails and attachments!

Indexing Tip #4: Update your indexes by telling dtSearch to add any new or changed documents, remove deleted documents and compress the updated index. This type of update tends to be much less time consuming than completely re-indexing. Even better, dtSearch can update its indexes automatically with no effect on ongoing concurrent searching. (**More**)

Indexing Tip #5: Check out general tips on optimizing indexing before you start a large index job. Following is just one example of the type of thing you need to know.

While search options like fuzzy searching are adjustable at search time, if you build a case and accent-sensitive index, the only way to change that setting is to rebuild the entire index again. With case and accent sensitive indexing on, your index size will be much larger, as your index will store *Frank*, *frank* and *FRANK* as separate words, instead of the same word. Worse, with case and accent-sensitive indexing on, a search for "*Frank Harvey*" would miss both "*frank harvey*" and "*FRANK HARVEY*." (**More**)