



DIGITAL DISCOVERY & E-EVIDENCE



VOL. 8, NO. 7 190-191

REPORT

JULY 1, 2008

Reproduced with permission from Digital Discovery & e-Evidence, Vol. 08, No. 07, 07/01/2008, pp. 190-191. Copyright © 2008 by The Bureau of National Affairs, Inc. (800-372-1033) <http://www.bna.com>

REVIEW AND PRODUCTION

De-duplication sounds so straightforward in theory: simply identify identical documents within a collection, and remove duplicates from the review and production workflow so that only one copy of each document remains. In reality, however, de-duplication is far more complex.

When the Devil is in the Duplicates

By ANGELA REEVES

Today's high-volume of Electronically Stored Information (ESI) encourages a proliferation of duplicate documents. For example, you may create a spreadsheet and save it to your local drive. Then, you may save it to a shared network drive for your colleagues to access. Finally, you may attach the spreadsheet to an e-mail and send it to a client for review. That night, the same spreadsheet in all three locations gets backed up to your company's backup systems. Without much thought, you've created multiple versions of the same document—all of which may be collected during discovery.

Angela Reeves is the manager of the Standards Department for Information Services at IE Discovery, a provider of Discovery Management services. She is responsible for the standardization and oversight of automated processes across all clients. Questions about this article may be directed to Reeves at AREeves@iediscovery.com.

Justifying Technical De-Duplication Solutions. While the volume of documents in a typical discovery grows ever larger, discovery time frames compress, leaving litigators with a significant increase in attorney review time and skyrocketing costs. This data explosion forces attorneys to seek ways to reduce volumes and reduce speed, while maintaining consistently high quality in the review process, even with multiple reviewers involved.

As demonstrated in *In re Priceline.Com Inc. Securities Litigation*, 233 F.R.D. 88 (D. Conn. 2005), courts are increasingly aware of the burden that duplicate documents can create, and they are forcing litigators to explain and defend decisions made during the discovery process. In the *Priceline.com* case, the plaintiffs moved to compel production of electronic data in discovery. The defendant possessed computer files that were not in an easily readable and searchable format, partly because they had been archived for backup purposes.

The court ordered the defendant to convert the files into PDF or TIFF format, eliminate all duplicate files, and produce a table containing metadata that allowed the plaintiff to search through and organize the files. Completing this process in a timely and efficient man-

ner would have been almost impossible without the assistance of an automated de-duplication solution.

In short, massively increasing volumes of data—combined with compressed review time lines and heightened awareness and expectations from the courts regarding duplicate documents—are leading litigators to embrace technical de-duplication solutions. De-duplication that is at least partially automated ultimately leads to greater consistency in the review process, lower risk of inadvertent disclosure, lower costs, and shortened review time frames. Soon, lawyers will be compelled to use technology to de-duplicate documents if they wish to stay abreast of changes in the technical and legal industries.

Defining Duplicates. While the meaning of “duplicate” may seem clear at first glance, documents that appear identical might not always be so. Even an extremely subtle difference between two documents, like the addition of a single comma, means that the documents are not identical and are not technically duplicates of each other. This can also be true of two documents with differing metadata (data about the document itself).

Total duplicates are document which match completely, and *near* duplicates are documents which have similarities, but are not total duplicates. There are three automated ways to identify total duplicates and near duplicates: matching of hash values, text comparison, and image comparison.

Matching of Hash Values. Most de-duplicating software identifies exact duplicates by utilizing a mathematical algorithm to produce a Hash value for each document, typically through an algorithm known as MD5Hash. The Hash value is like the document’s fingerprint—the Hash process analyzes the document and assigns a 32-character value based on the document’s electronic content.

Once a Hash value is assigned to every document in a collection, exact duplicates with matching Hash values can be removed from further review or production. This automated process of identifying exact duplicates saves the discovery team from manually examining every document to determine whether it is a duplicate.

Text and Image Comparison. Other de-duplication processes run comparisons of document text or images. Text-based comparisons give a percentage match of the content between two documents. The text can be from Optical Character Recognition (OCR) if the document was scanned from paper, or it can be extracted from the searchable text in an electronic document.

Image-based comparisons also give percentage matches between documents based on a mathematical comparison of the document’s colors and shapes. Image-based de-duplication is rarer and less tested, since the technology to compare images has only recently been developed. Either of these technologies can be used to find total duplicates or near duplicates.

An example illustrates the difference between a Hash comparison and a text-based comparison: one file is a PDF file, and another file is the Word document used to create the PDF. Hashing technology alone will not identify these documents as duplicates because they are in different formats, and therefore will have different digital fingerprints. However, those documents would have identical text content, and therefore a text-based de-

duplication process would identify them as total duplicates.

Additional Concerns. While it is important to identify exact duplicates as a first step in the de-duplication process, identifying near-duplicates, with a match percentage of less than 100 percent, is just as critical and far more complex. Since it is impossible to tell the percentage difference between two documents based on their Hash values, near-duplicates can only be identified through text-based or image-based comparison tools.

The reasons for using de-duplication vary widely from case to case; therefore, guidelines for setting a match percentage threshold also vary widely. For example, in a contracts case where all the documents are electronic, a match percentage of 99 percent would still need to be verified by human review, because a 1 percent difference could be a change in a contract amount, which would be a substantive differentiation.

However, when de-duplication is used to speed pre-hearing review of a production from opposing counsel, the attorney may choose to group documents together with similarities of at least 80 percent, only review one document per group, and still have a high level of confidence that the vast majority of the content in the collection has been seen.

When setting a match percentage threshold for a collection of documents, it is important to consider the probability of inadvertent disclosure, or of withholding relevant documents. Although there has not been a known case of someone having discarded the “smoking gun” accidentally by relying completely on automation for identifying near-duplicates, it is not hard to think of scenarios where choosing the wrong draft of a document could have negative consequences for the integrity of the case.

A variation of content-based de-duplication is e-mail thread analysis, which identifies earlier e-mails in a thread which are completely contained within later e-mails. This allows the earlier e-mail to drop out of reviews and productions, because the content is intact in the final, most complete e-mail. In the case of e-mail thread analysis, the overall percentage match of the two e-mails is not important; it only matters that the earlier e-mail is thoroughly represented in the later e-mail.

When to De-Duplicate. There are several points during the discovery process when files can be de-duplicated. They include:

- **Collection:** When de-duplication is conducted during collection, it can only be performed on electronic files as they are collected. This means that compressed files (such as “ZIP” files) will not be de-duplication candidates during the collection process since Hash values or percentage matches cannot be obtained until the contents are extracted. De-duplication during collection may be acceptable for document collections with a single custodian, to ensure that each document from that custodian is only collected once.

- **Processing:** Processing of electronic files includes extracting the contents of compressed or ZIP files, assigning Hash values, and possibly converting documents to images and text. Processing of paper documents involves scanning the paper, and may include sending documents through OCR to provide searchable text. Near-duplicates are difficult to determine until processing has been completed, but identifying total duplicates is very reliable during processing.

■ **Post-Processing:** Post-processing of documents can be used to reliably identify near-duplicates, as well as total duplicates. By this point in the process, litigators should have clear-cut definitions of duplicates and near-duplicates for the individual case, and the collection is ready for human review.

Like any discovery process which could be challenged, one key to a successful and defensible imple-

mentation of de-duplication is to keep a comprehensive record of the decision-making processes that led to the inclusion or exclusion of each file. This record, along with the approaches described above, should result in a successful de-duplication effort and lead to a more consistent and cost-effective review and production process.