

HASH: THE NEW BATES STAMP

*Ralph C. Losey**

Originally Published:
12 *Journal of Technology Law & Policy* 1 (June 2007)

I.	INTRODUCTION	2
II.	BATES STAMPING	4
	<i>A. Origins</i>	4
	<i>B. Evolution</i>	6
	<i>C. Modern Use</i>	7
III.	INADEQUACIES OF THE BATES STAMP IN THE TWENTY-FIRST CENTURY	9
IV.	THE HASH ALGORITHM	12
	<i>A. Digital Fingerprint of All ESI</i>	12
	<i>B. Process of Hashing</i>	13
	<i>C. Types of Hash</i>	13

* J.D. *cum laude*, 1979, University of Florida College of Law; B.A., 1973, Vanderbilt University. The author is a practicing attorney and Co-Chair of Akerman Senterfitt's electronic discovery practice group. See Akerman Senterfitt, e-Discovery, at <http://www.akerman.com/public/practice/pDescr.asp?id=140>. The author was the first attorney in Florida with a web site, and has been involved with computers and technology since 1978. He currently maintains two law-related web sites and a blog on e-discovery. Ralph Losey's Law Web Site, at <http://www.FloridaLawFirm.com>; e-Discoveryteam Web Site, at <http://www.e-DiscoveryTeam.com>; e-Discoveryteam Blog, at <http://ralphlosey.wordpress.com>. Research assistance is gratefully acknowledged from Adam C. Losey, his son, now a student at the University of Florida College of Law, and Kelly Garcia, associate attorney at Akerman Senterfitt.

D. <i>Examples of Hash</i>	16
E. <i>The Irreversibility of Hash</i>	17
F. <i>The Value of Irreversibility in e-Discovery</i>	18
V. THE APPLICATION OF HASH TO AUTHENTICATE ESI	20
A. <i>The Special Importance of Hash in Native File Productions</i>	21
B. <i>Hash is Widely Accepted in Civil Cases</i>	23
C. <i>Hash is also Widely Used in Criminal Cases</i>	27
D. <i>Commercial and Governmental Uses of Hash</i>	29
E. <i>Electronic Data Transfers</i>	30
F. <i>Federal Court Filings</i>	32
G. <i>Peer-to-Peer Transfers</i>	33
VI. THE APPLICATION OF HASH TO FILTER ESI.....	35
A. <i>De-Duplication</i>	35
B. <i>Known ESI Elimination</i>	39
VII. A MODEST PROPOSAL	39

I. INTRODUCTION

For over one hundred years, complex litigation has relied upon the ubiquitous Bates stamp to try and maintain order and clarity in paper evidence by placing sequential numbers on documents. In today's world of vast quantities of electronic documents, the days of the Bates stamp are numbered. Instead, the future belongs to a new technology, a computer-based mathematical process known as "hash." The hash algorithm analyzes a computer file and calculates a unique identifying number for it, called a hash value. No two electronic records have the same hash value. For that reason, it is called the "digital fingerprint" of electronic documents. Here is an example of a hash value: 162B6274FFEE2E5BD96403E772125A35. Unlike a Bates stamp, the hash value of a file will automatically and necessarily change if the file is altered. Thus, hash can both provide objective order and authenticate an unlimited number of electronic documents. For these and other reasons, the author proposes a new electronic-evidence naming protocol be adopted based on algorithmic hash values, instead of sequential numbers. As explained more fully in the conclusion, the proposal truncates the full alphanumeric hash value of electronic documents to the first and last three

numbers, and so the above hash value is shortened to the more manageable sequence: 162.A35.

This Article begins by tracing the history of the Bates stamp, how it has been used in past litigation, and why it is inadequate to meet the challenges and unique problems of today's technological world. Next, hash and hashing will be explained with as little mathematics as possible. Some of the remarkable qualities of hash will be examined and a few of the many uses of hash in law and society will be described. As will be shown, hashing has many advantages over Bates stamping, including authentication, filtering, and unique search capabilities, and so is ideally suited to meet the litigation challenges of today and tomorrow. Only hash, hashing, and hash marking can cope with the incredible volume of data generated today and protect the integrity of evidence and the judicial system. As evidence shifts from paper documents to electronically stored information in ever-increasing quantities, the legal profession must necessarily adopt hash values over simple Bates stamping.

This Article includes all significant cases to date that mention or relate to hash. Although the work is, in this sense, intended to be a complete legal reference, no doubt the comprehensive quality of this Article will necessarily be short-lived. The number of cases mentioning hash increases every month. This corresponds with the many new and innovative applications developed for hash in society at large. Although this Article will be quickly dated as a complete reference, it should serve as a good building block for future articles on hash and the law. In the meantime, this Article may be of some small help to other lawyers and jurists, who may struggle to understand the many elusive qualities and applications of this profound mathematical algorithm.

The proposal made at the conclusion to adopt a new hash-based naming system will, it is hoped, be of more long-lasting value, and will create a new standard for electronic document identification. I urge the legal community and electronic discovery industry to begin using this new naming protocol as soon as possible. The truncated hash value system proposed here is a timely replacement to the archaic Bates stamp. The naming protocol is simple, practical, and easy to use. Most importantly, it can be immediately employed with today's technology to avoid needless confusion and disputes concerning the identification and authenticity of electronic documents. In this way, the proposed hash protocol can help ensure the progress and integrity of our legal system in the dawning new age of electronic discovery.

II. BATES STAMPING

A. Origins

Edwin G. Bates invented the Bates numbering machine, and the Bates Manufacturing Company patented it in 1893.¹ Although the appearance of the Bates numbering machine has changed over the last hundred years, its basic function and method of operation remain the same.² A Bates machine uses a self-inking stamp and a mechanically advancing sequence of numbers. Each time the handle of the machine is pressed, a number is imprinted on the document below. With every press of the handle, the number advances sequentially and the next number is inked onto the document.

Thomas Edison knew a good invention when he saw it, so he bought the Bates Company.³ Edison's Bates brand so dominated the automatic-hand-held-numbering-machine market that numbers imprinted on multi-page documents were eventually known as Bates numbers.

Incredibly, Bates stamping machines are still used in nearly all law firms in the United States.⁴ The only exception may be a few small firms (the author has encountered them) that never add page numbers to anything, or if they do, handwrite the numbers on the pages. Even technologically advanced law offices occasionally use manual Bates stamping machines to put page numbers on a small set of documents. The Bates number appearing on the first page of the document identifies a particular document in a set of documents. This significantly aids in document identification and organization, especially when there are many hundreds or thousands of pages involved. Each page in a set of documents has its own unique number. The number labeling also makes it easier to prove that documents have been produced to one party from another.

1. U.S. Patent No. 489,449 (filed Oct. 2, 1891); *see also* Christopher L.T. Brown, Bates Numbering—What's in a Number Anyway?, Technology Pathways, Technical White Paper (July 17, 2003), available at <http://www.techpathways.com/uploads/BatesNumbering.pdf> (last visited May 24, 2007).

2. *See* Early Office Museum Web Site, Antique Date, Time, Number, & Name Stamps, available at <http://www.officemuseum.com/stamps.htm> (last visited May 24, 2007) (containing a drawing of the original Bates Automatic Numbering Machine from an 1897 advertisement); *see also* U.S. Patent No. 489,449 fig.1 (filed Oct. 2, 1891).

3. Rutgers University, *Edison Papers: Company Records Series—Bates Manufacturing Company*, available at <http://edison.rutgers.edu/NamesSearch/glocpage.php3?gloc=CK300&> (last visited May 24, 2007).

4. This is based on the author's experience as a practicing attorney for the last twenty-seven years.

Absent some unusual circumstances, all American courts accept this system of identification (coupled with actual custodian testimony where there is no stipulation) as sufficient identification and authentication to allow documents to be admitted into evidence.⁵ In *United States v. Block*, the court references the ubiquitous Bates numbering system and found it adequate to authenticate the document for admission into evidence as follows:

She stated that she could identify the location from which the individual documents were seized by referring to the Bates stamp and property receipt. Furthermore, Van Etten explained the Bates stamping procedure used by his company, by which the documents inside each box received a stamp with a prefix specifying the box in which they had been placed, and were numbered consecutively. Because this testimony is sufficient to support a finding that the documents in question are what the government claims—documents seized from Block’s desk during execution of the search warrant at NWE—the requirement of authentication for the admissibility of this evidence is satisfied.⁶

Of course, a lawyer actually has to put the documents into evidence, and it is not enough to simply refer to their Bates numbers. This is exactly what the plaintiffs did in *Andretti v. Borla Performance Industries, Inc.*⁷ Plaintiffs referred to documents by their identifying Bates stamp numbers, but never actually introduced the documents themselves into evidence.⁸ These documents were needed for the plaintiffs to prove their damages. When plaintiffs learned of this error they argued that since both parties knew what documents were intended by the Bates stamp number references, this alone was sufficient to support the inclusion of the documents into the record.⁹

The court disagreed, holding that reference to documents merely by Bates numbering was not sufficient, even if both parties knew what was intended.¹⁰ The court held that the documents themselves had to be placed

5. See, e.g., *United States v. Block*, 148 F.App’x 904, 911 (11th Cir. 2005), *cert. denied*, (126 S. Ct. 1175) (Jan. 23, 2006).

6. *Id.* at 911.

7. 426 F.3d 824, 831 (6th Cir. 2005).

8. *Id.*

9. *Id.*

10. *Id.*

in evidence in order to support the claim for damages.¹¹ The district court noted it “could not consider the evidentiary value of documents of which it was not aware.”¹²

As most trial attorneys know from experience, one of the failings of Bates stamping is that a single document often has several different Bates stamp numbers. This occurs frequently because, in any fairly large document production, documents usually appear multiple times. Consequently, a single document receives several different numbers. Bates stamps are often confusing for that reason.

However, the alternative of not Bates stamping documents is even worse. Locating and identifying a particular document in a large document production can be time-consuming and confusing. Again, this is something that any practicing attorney knows all too well. This problem is particularly difficult when there are only subtle differences between documents, and only close inspection reveals these differences.

Although no one knows when the first lawyer used a Bates machine to affix numbers to stacks of documents,¹³ the author recalls that Bates stamping was still infrequently used when he started practice in 1979. Because most cases did not involve more than a few hundred pages of documents, there was no pressing need to have special numbers applied to keep track of them. Lawyers would identify documents by their names, dates, and parties. In multi-page documents, the internal pagination of the particular document would be used. In cases where there were many documents, and no Bates stamping, problems quickly developed when documents had the same name, there was an exceptionally large number of documents, or reference was made to long documents without page numbers.

B. Evolution

In the author’s experience, a rapid increase in the number of lawsuits involving thousands of pages of evidence took place in the 1980s. Word processors were prevalent, and businesses and law firms went from “mag-card” electric typewriters to the first personal computers. The number of documents involved in typical disputes began to multiply, leading to a greater need to use the Bates stamp machine to help keep track of them all.

11. *Id.*

12. *Andretti*, 426 F.3d at 831.

13. It may well have been Frank Lewis Dyer of New York, Thomas Edison’s attorney, who served as the president of the Bates Manufacturing Company after Edison acquired it. See National Park Service Web Site, available at http://www.nps.gov/archive/edis/edifun/edifun_hschoo/other_muckers.htm (last visited May 24, 2007).

The early promise of a paperless office was never realized. Rather, the opposite occurred; businesses began churning out ever-increasing amounts of documents as it became easier to do so.

Eventually, in the 1990s, clever legal secretaries trying to cope with the tedium of manual Bates stamping of thousands of documents devised a way to use computers to print Bates numbers onto stick-on labels. The secretaries could then peel off the labels and stick them onto each page. Many lawyers and law firms still do this. Although only slightly less tedious a chore than hand-stamping numbers, this computer enhancement allowed Bates stamps to easily include letters and words, usually names, as a prefix before the consecutive numbers. This alphanumeric feature was much appreciated by beleaguered lawyers trying to keep track of thousands of pages of documents with merely the numbers as a guide.

With this innovation, documents produced by a witness could be Bates labeled with his or her name. For instance, a thousand pages of medical records produced by a Dr. Smith could be Bates labeled “Dr. Smith 0001 - Dr. Smith 1000.” At a deposition, hearing, or trial, a lawyer could easily identify a particular document within an entire production. For example, a lawyer would say “the MRI report of April 7th 1997 Bates stamped ‘Dr. Smith 0075.’” Everyone could quickly find that document. Moreover, it would be clear on the record of the proceedings precisely to what document and to which version of the MRI report the lawyer was referring.

C. Modern Use

The next stage in the evolution of Bates stamping came when paper documents were scanned into Tagged Image File Format (TIFF), or Adobe Acrobat Portable Document Format (PDF) files. Finally, the computer alone could directly add Bates numbers. This advance freed secretaries and paralegals from the tedious task of hand-marking each page.

The latest change in Bates stamping, which may be its dying gasp, has been adopted by all electronic discovery (e-discovery) vendors. In fact, most e-discovery software programs currently include a service or feature where electronic documents are converted into photo image TIFF or PDF files and a Bates number is electronically added to each page of the file. Sometimes, the Bates number is simply added to the computer file names.¹⁴ Thus, when a computer file containing 10,000 pages of records

14. See, e.g., Brown, *supra* note 1; Bates Numbering: Black Ice Printer Drivers, Black Ice Software, at <http://www.blackice.com/Printer%20Drivers/Bates%20Numbering.htm> (last visited May 24, 2007).

is viewed or printed, each page will have a number 1 to 10,000 added to it, or in the alternate file naming protocol, a sequential Bates number will be added to the original file names. There are obvious advantages to such a sequential numbering system, including ease of use, at least with small numbers of documents, and familiarity. But in cases involving voluminous amounts of electronic documents, these advantages disappear. The conversion to image files to allow Bates stamping strips an electronic document of functionality. Moreover, since Bates stamping was designed for paper, not dynamic computer files, it provides a poor naming and authentication protocol for electronic documents.

A recent case in district court in Illinois reached the same conclusions.¹⁵ The defendant converted e-mails into TIFF files, which eliminated most of their metadata.¹⁶ One justification for the conversion was to enable the defendant to “add Bates numbers to every page of every document, thereby making it possible for . . . [the plaintiff] to quickly and efficiently locate and authenticate any documents that Plaintiff refers to or relies upon as this lawsuit goes forward.”¹⁷ The court was not persuaded by the defendant’s argument, holding that the benefits of adding Bates numbers to the TIFF documents did not justify the failure to produce the

15. *Hagenbuch v. 3B6 Sistemi Elettronici Industriali S.R.L.*, No. 04-C-3 109, 2006 WL 665005, at *4 (N.D. Ill. Mar. 8, 2006).

16. *Id.* at *2. “Metadata” literally means “data about data” and is one of the key concepts in electronic discovery today. *See, e.g.*, CATHERINE SANDERS REACH, NEUMILLER & BEARDSLEE, METADATA (AND OTHER THINGS THAT GO BUMP IN THE NIGHT) (ABA Legal Technology Resource Center, July 27, 2006), available at <http://www.abanet.org/tech/ltrc/presentations/neumillermetadata.pdf> (last visited May 24, 2007). All computer files have metadata embedded or associated with them that provide information about the files. For instance, e-mail software embeds in e-mail files information about its author, creation date, attachments, and identities of all recipients, including those who received a cc or bcc. The printout of an e-mail, which is essentially a TIFF version of the e-mail, may not show the blind copies. The metadata will also maintain the history of an e-mail, its conversation thread, such as who replied, who forwarded, the folder in which it was filed, and even when or if an e-mail was opened. Also, when e-mail is used to transmit documents as attachments, which is very common today, the e-mail metadata allows you to know which documents were attached to which e-mails. The printout of an e-mail, which is essentially a TIFF version of the e-mail, will not show any of this metadata. SCOTT NAGEL, EMBEDDED INFORMATION IN ELECTRONIC DOCUMENTS: WHY METADATA MATTERS (Law Practice Today, ABA Law Practice Management Section, July 2004). Metadata is currently the topic of ethical debate, especially in the context of inadvertent disclosures of confidential information. *See* Formal Opinion 06-442, *Review and Use of Metadata* (ABA Standing Committee on Ethics and Professional Responsibility, Aug. 5, 2006), available at http://www.pdfforallawyers.com/files/06_442.pdf (last visited May 24, 2007); David Hricik & Robert B. Jueneman, *The Transmission and Receipt of Invisible Confidential Information*, 15 PROF. LAW. 18 (2004).

17. *Hagenbuch*, No. 04-C-3 109, 2006 WL 665005, at *2-3.

designated electronic media.¹⁸ The court observed that there were various ways the parties could identify electronic records without resorting to Bates numbering, including, “relying on file names and page numbers.”¹⁹ The court also rejected the argument that TIFF documents and Bates stamping should be used to make evidence tampering more difficult and easier to detect because “the mere fact that a document is in TIFF format and Bates stamped does not make it impossible for a party to tamper with the contents of the document.”²⁰ The court was “confident that both parties will be double checking the authenticity of any documents relied upon by the other side and, while Bates stamping may provide a simple method for locating and authenticating documents, it is certainly not the only method.”²¹

III. INADEQUACIES OF THE BATES STAMP IN THE TWENTY-FIRST CENTURY

Part of the problem facing litigators today is the sheer volume of “electronically stored information”²² (ESI) now involved in many lawsuits.

18. *Id.* at *3-4.

19. *Id.* at *4.

20. *Id.*

21. *Id.*

22. Electronically Stored Information is the terminology utilized in the Federal Rules of Civil Procedure as revised effective December 1, 2006 to signify electronic records of all types, including, but not limited to computer files. FED. R. CIV. P. 34(a). Neither the amendments nor the accompanying Committee Notes define the phrase, but it is commonly “understood to mean information created, manipulated, communicated, stored, and best utilized in digital form, requiring the use of computer hardware and software.” Kenneth J. Withers, *Electronically Stored Information: The December 2006 Amendments to the Federal Rules of Civil Procedure*, 4 NW. J. TECH. & INTELL. PROP. 171, ¶ 9 (2006), available at <http://www.law.northwestern.edu/journals/njtip/v4/n2/3> (last visited May 29, 2007). ESI is intended to be broadly construed to cover the known formats of electronic documents, and the as yet unknown forms of data that will certainly arise in the future. Rule 34(a) specifically states that the scope of the production includes any “other data or data compilations stored in any medium from which information can be obtained.” *Id.* ¶¶ 73-76. The Committee Notes accompanying the proposed amendments to Rule 34 explain why the term is not precisely defined:

The wide variety of computer systems currently in use, and the rapidity of technological change, counsel against a limiting or precise definition of electronically stored information. Rule 34(a)(1) is expansive and includes any type of information that is stored electronically. A common example often sought in discovery is electronic communications, such as e-mail. The rule covers—either as documents or as electronically stored information—information “stored in any medium,” to encompass future developments in computer technology. Rule

This volume is a direct result of the use of computer technology in business. The amount of ESI created each year is astounding and almost defies imagination. A landmark study by the School of Information Management and Systems of the University of California at Berkeley²³ estimated that more than 99% of information created and stored in 2001 was electronic. Moreover, researchers estimate that in 2002 alone, about five exabytes of new information were created worldwide. That may not sound like much, until you discover the size of an exabyte. Five exabytes equals *all of the words ever spoken by human beings*.²⁴ The Berkeley study explains it this way:

How big is five exabytes? If digitized with full formatting, the seventeen million books in the Library of Congress contain about 136 terabytes of information; five exabytes of information is equivalent in size to the information contained in 37,000 new libraries the size of the Library of Congress book collections. . . . The world population is 6.3 billion, thus almost 800 MB of recorded information is produced per person each year. It would take about 30 feet of books to store the equivalent of 800 MB of information on paper.²⁵

The Berkeley study further estimated that in 2003 the world sent 31 billion e-mails and 5 billion instant messages a day.²⁶ It estimated that the

34(a)(1) is intended to be broad enough to cover all current types of computer-based information, and flexible enough to encompass future changes and developments.

2006 Amendments with Committee Notes, Rule 34(a), available at http://www.uscourts.gov/rules/EDiscovery_w_Notes.pdf (last visited May 24, 2007).

23. Peter Lyman & Hal R. Varian, How Much Information (2003), available at <http://www.sims.berkeley.edu/research/projects/how-much-info-2003/execsum.htm> (last visited May 24, 2007).

24. One exabyte equals 1,000,000,000,000,000 bytes OR 1,018 bytes. Two exabytes equal the total volume of information generated in 1999. *Id.* See also [Whatis.com](http://whatis.com), How many bytes for?, available at http://searchstorage.techtarget.com/sDefinition/0..sid5_gci944596.00.html (last visited May 24, 2007). “Exabyte” is derived from the Greek word for “beyond” or “outside.”

DOUGLAS DOWNING ET AL., DICTIONARY OF COMPUTER AND INTERNET TERMS (9th ed. 2006).

25. Lyman & Varian, *supra* note 23.

26. *Id.*

overall e-data was increasing by 30% or more per year. As of 2007, the world sends an estimated 100 billion e-mails per day.²⁷

A more recent study in 2007 estimates that in 2006 the world created 161 exabytes of data.²⁸ This is explained to be “about 3 million times the information in all the books ever written.”²⁹ The study projects that by 2010 the amount of ESI added annually will increase more than six-fold from 161 exabytes to 988 exabytes.

While these estimates may be hard to believe, the law is quickly discovering that the information explosion is no myth; it is a harsh reality.³⁰ For example, when Enron collapsed and all of its records became the subject of government investigation and numerous lawsuits, the parties discovered that this company alone maintained digital evidence over 200 terabytes³¹ in size.³² Comparing this number to the size of the entire print collection of the Library of Congress, Enron had twenty times more ESI than the library. From this, we may reasonably infer that, by the turn of the century, most major corporations in the United States had already stored enough ESI to fill twenty Libraries of Congress.

Sequential Bates stamping is inadequate when you start dealing with these kinds of numbers.³³ The information explosion is challenging all

27. George L. Paul & Jason R. Baron, *Information Inflation: Can the Legal System Adapt?*, 13 RICH. J.L. & TECH. 10, 18 (2007), at <http://law.richmond.edu/jolt/v13i3/article10.pdf> (last visited May 24, 2007).

28. John F. Gantz, *The Expanding Digital Universe: A Forecast of Worldwide Information Growth Through 2010*, at 1, IDC White Paper (Mar. 2007), at http://www.emc.com/about/destination/digital_universe/ (last visited May 24, 2007).

29. *Id.*

30. See GEORGE L. PAUL & BRUCE H. NEARON, *THE DISCOVERY REVOLUTION: E-DISCOVERY AMENDMENTS TO THE FEDERAL RULES OF CIVIL PROCEDURE 4* (ABA Publishing 2006) (“Organizations have thousands if not tens of thousands of times as much information within their boundaries as they did 20 years ago.”).

31. Tera is a metric prefix meaning one trillion (1,000,000,000,000) (10 to the 12th power). In computer memory, one terabyte equals approximately one trillion (1,000,000,000,000) bytes, or to be exact, 1,099,511,627,776 bytes. It can also be expressed as 2 to the 40th power, or 1024 to the 4th power, and is equal to 1,000 gigabytes. It is derived from the Greek word for “monster” or “freak.” DOWNING ET AL., *supra* note 24.

32. This is according to Craig Ball, a lawyer and computer forensics expert retained by the plaintiffs in the Enron cases. Craig Ball, 5 on EDD: Five Articles on Electronic Data Discovery, available at http://www.utahbar.org/cle/fallforum/materials/general2/five_of_electronic_discovery.pdf (last visited May 24, 2007).

33. The tobacco litigation cases are a good example of this problem. There were millions of documents and keeping track of them all was very difficult. To see how lawyers tried to do this and to get an idea of the complexity of the problem, see Tobacco Institute Index to Documents, available at <http://www.tobaccoinstitute.com/navindex.asp>; Bates Numbers, available at <http://tobacco.health.usyd.edu.au/site/gateway/docs/pdf/Bates.pdf> (last visited May 24, 2007).

aspects of the legal profession to the core, not just document identification.³⁴ Indeed, our whole culture seems to be going through fundamental change driven by advances in technology and writing that some scholars believe heralds an entirely new phase of civilization.³⁵

IV. THE HASH ALGORITHM

A. Digital Fingerprint of All ESI

What is hash? As the term is used today in electronic discovery, it is neither a food nor an illegal substance; hash is a mathematical process. To be precise, hash is an encryption algorithm. Hashing generates a unique alphanumeric value to identify the total combination of bits and bytes that make up a particular computer file, group of files, or even an entire hard drive.³⁶ The unique number of a computer file is its hash value, also known in mathematical parlance as the “condensed representation” or “message digest” of the original message.³⁷ It is more popularly known today as a “digital fingerprint.”³⁸

34. Paul & Baron, *supra* note 27, at 1-15.

35. *Id.* at 1-10.

36. The full hard drive hashing process is explained in *Sanders v. State*, a child pornography case:

Lee explained that when he takes a hard drive from a computer, he uses a program like EnCase to automate the task of searching and finding the files on it. An image of the drive is taken; the files are copied, and EnCase validates the copy by an “MD5 hash,” a 128-bit algorithm that verifies the image. The MD5 hash is essentially a “digital fingerprint” of a drive, and if the hash values match, Lee said that “basically there’s no chance” that an error occurred in making an exact duplicate of the original computer file. Lee used EnCase on computer files taken from Sanders’s computer. EnCase indexed the files, and Sanders was able to retrieve deleted files containing child pornography from Sanders’s computer.

Sanders v. State, 191 S.W.3d 272, 278 (Tex. App. 2006).

37. For more technical information on the mathematics of hash, see Ronald Rivest, RFC 1321, The MD5 Message-Digest Algorithm, available at <http://asg.web.cmu.edu/rfc/rfc1321.html> (last visited May 24, 2007); MD5 Homepage (Unofficial), available at <http://userpages.umbc.edu/~mabzug1/cs/md5/md5.html> [hereinafter RFC 1321] (last visited May 24, 2007); Tim Boland & Gary Fisher, Selection of Hashing Algorithms, available at <http://www.nslr.nist.gov/documents/hash-selection.doc> (last visited May 24, 2007).

38. The Sedona Conference Glossary: E-Discovery & Digital Information Management, The Sedona Conference Working Group Series, May 2005, at 21 [hereinafter Sedona Conference Glossary] (defining “Hash” as “a mathematical algorithm that represents a unique value for a given

B. Process of Hashing

It is important to understand that the computational process of hashing is lightning fast in execution. For example, if all of the printed materials in the Library of Congress, which is estimated to equal 136 terabytes of data,³⁹ were in electronic format, they could all be hashed in a matter of hours, if not minutes.⁴⁰ Manual Bates stamping of each page would, by comparison, take several lifetimes of work.

Technically, hashing is based on the substitution and transposition of data by various mathematical formulas. Thus, the process is called “hashing,” in the linguistic sense of “to chop and mix.” Hashing is a formula, which, so to speak, allows you to boil a file down to an essential number. The hash value is commonly represented as a short string of random-looking letters and numbers, which are actually binary data written in hexadecimal notation. The hash value is commonly called a file’s “fingerprint” because it represents its absolute uniqueness. If two computer files are identical, they will have the same hash value. Even if the files have a different name, if their contents are exactly the same, they will have the same hash value. But if you simply change a single comma in a thousand page text, that document will have a completely different hash number than the original. There are no similarities in the hash numbers based on similarities in the files. Each number is unique.

C. Types of Hash

Many kinds of effective hash formulas have been invented, but two are widely used today: the SHA- 1 and MD5 algorithms. The Secure Hash

set of data, similar to a digital fingerprint.” It defines “Hashing” or “Hash Coding” as a method “to create a digital fingerprint that represents the binary content of a file unique to every electronically-generated document; assists in subsequently ensuring that data has not been modified.”), *available at* <http://www.capitallegals.com/Pdf/Glossary.pdf> (last visited May 24, 2007).

39. Lyman & Varian, *supra* note 23.

40. Assuming the library books were together in one large file, instead of millions of smaller files, and assuming a cryptographic algorithmic speed of one hundred megabytes (100,000,000,000) (109) per second, which seems reasonable on a high speed CPU today based upon crypto benchmarks. Speed Comparison of Popular Crypto Algorithms, Crypto++ 5.2.1 Benchmarks, *available at* <http://www.eskimo.com/~weidai/benchmarks.html> (last visited May 24, 2007). In theory, you could hash ten terabytes (10,000,000,000,000) (1012) in only 100 seconds. Theory confirmed in private correspondence with hash and cryptology expert, Bruce Schneier. See E-mails to and from Schneier to Losey, Nov. 17, 2006 (on file with author); *see also* [Schneier.com](http://www.schneier.com) Web Site, *available at* <http://www.schneier.com/> (last visited May 24, 2007); Schneider on Security: NIST Hash Workshop Liveblogging, *at* http://www.schneier.com/blog/archives/2005/11/nist_hash_works_4.html (last visited May 24, 2007).

Algorithm (SHA) was originally developed by the National Institute of Standards and Technology (NIST) at the U.S. Department of Commerce.⁴¹ SHA-1 is an improved revision to the original SHA version and was published in 1994. SHA-1 produces a 160-bit (20 byte) file digest. Although slower than MD5, this larger digest size makes it even more reliable, and more effective in a cryptology code-breaking context.⁴²

MD5, or Message Digest 5, was developed and published by Professor Ronald L. Rivest of the Massachusetts Institute of Technology in 1992.⁴³ Rivest describes his algorithm as follows:

The algorithm takes as input a message of arbitrary length and produces as output a 128-bit “fingerprint” or “message digest” of the input. It is conjectured that it is computationally infeasible to produce two messages having the same message digest, or to produce any message having a given prespecified target message digest. The MD5 algorithm is intended for digital signature applications, where a large file must be “compressed” in a secure manner before being encrypted with a private (secret) key under a public-key cryptosystem such as RSA.⁴⁴

The 128 bit (16 byte) message digest of MD5, as compared to the 160-bit (20 byte) of SHA- 1, makes it a faster implementation than SHA- 1. MD5 ’s speed, coupled with its continued reliability, makes it a commonly used hash algorithm in computer forensics.⁴⁵

Both are very effective because, according to mathematicians, it is “computationally infeasible,” in other words, impossible, for two different

41. NIST, Announcing the Standard for Secure Hash Standard, FIPS Pub. Doc. 180-1 (Apr. 17, 1995), available at <http://www.itl.nist.gov/fipspubs/fip180-1.htm> (last visited May 24, 2007).

42. Note that the NIST is working on even larger and stronger hash algorithms for national security cryptology purposes, such as SHA-512, and plan to phase out SHA-1 as the national standard in 2010. NIST, NIST Brief Comments on Recent Cryptoanalytic Attacks on Secure Hashing Functions and the Continued Security Provided by SHA-1 (Aug. 25, 2004) [hereinafter NIST Brief Comments], available at http://csrc.nist.gov/hash_standards_comments.pdf (last visited May 24, 2007).

43. Rivest, *supra* note 37; see also Web Page on Rivest at the Cryptographers’s Lounge Web Site, at <http://www.cryptolounge.org/wiki/MD5>; Fast-Sum Software Company’s Web Site, at <http://www.fastsum.com/support/md5-checksum-utility-faq/md5-hash.php> [hereinafter Fast-Sum Software] (explaining the basis of its company’s product in MD5 hash); Secure Hash Algorithm Directory, at <http://www.secure-hash-algorithm-md5-sha-1.co.uk/index.htm>.

44. Rivest, *supra* note 37, Executive Summary, at 1 n.23.

45. See Shawn McCreight & John Patzakis, Guidance Software, Hash Sets and Their Proper Construction, at http://www.guidancesoftware.com/support/downloads/hashsets/hashsets_wp.pdf (last visited May 24, 2007).

files to produce the same hash value.⁴⁶ An MD-5 hash can generate more than 340,000,000,000,000,000,000,000,000 (that is 340 billion, billion, billion, billion) possible values. The SHA- 1 algorithm generates a range of values over four billion times larger than MD5.⁴⁷ Therefore, even though there is a finite number of possible hash values, and, in theory at least, an infinite number of possible data inputs, the odds of two different files generating the same hash value (called a “collision” in the language of cryptanalysts) are “computationally infeasible.”⁴⁸ Reports as to artificially created collisions are theoretical math exercises performed on high-speed computers, and at present, collisions do not pose any real threats to the integrity of hash.⁴⁹

This means that the hash value of two different e-documents must always be different. The odds of a coincidental “collision” where different files have the same hash is “computationally infeasible,” on the order of 1 in 100 million, million, million, million, million, million.⁵⁰ Since such a large number is hard to comprehend, it is sometimes instead said that the odds of a collision are “less than one in one billion.”⁵¹

46. See BRUCE SCHNEIER, APPLIED CRYPTOGRAPHY: PROTOCOLS, ALGORITHMS, AND SOURCE CODE IN C 30 (2d ed. 1996).

47. *Id.*

48. Rivest, *supra* note 37.

49. See, e.g., NIST Brief Comments, *supra* note 42; Thomas C. Greene, *Crypto Researchers Break SHA-1*, REGISTER, Feb. 17, 2005, available at http://www.theregister.com/2005/02/17/sha1_hashing_broken/ (last visited May 24, 2007). *But see* John Kelsey & Tadayoshi Kohno, *Herding Hash Functions and the Nostradamus Attack—DRAFT*, available at http://cyphunk.files.wordpress.com/2006/02/HerdingHash_paper.pdf (last visited May 24, 2007); Magnsu Daum & Stefan Lucks, *Attacking Hash Functions by Poisoned Messages: The Story of Alice and her Boss*, Institute for Cryptology and IT-Security, Ruhr-Universitat Bochum, Germany, at <http://www.cits.rub.de/MD5Collisions/> (last visited May 24, 2007). According to unconfirmed Chinese press reports, the U.S. government will stop using SHA-1 in four years and adopt a new more advanced hash algorithm because associate professor Wang Xiaoyun of Beijing’s Tsinghua University and Shandong University of Technology, and her associates, have recently cracked SHA-1. *Chinese Professor Cracks Fifth Data Security Algorithm*, EPOCH TIMES, Jan. 11, 2007, available at <http://en.epochtimes.com/news/7-1-11/50336.html> (last visited May 24, 2007).

50. RONALD A. GOVE ET AL., DATA SECURITY AND PRIVACY LAW, Technical Privacy Measures: Encryption, ch. 4, § 32 (Nat’l Bus. Inst. Mar. 2007).

51. See FEDERAL JUDICIAL CENTER, MANAGING DISCOVERY OF ELECTRONIC INFORMATION: A POCKET GUIDE FOR JUDGES 24 (2007) [hereinafter POCKET GUIDE FOR JUDGES]. This *Pocket Guide for Judges* explains a “hash value” as follows:

A unique numerical identifier that can be assigned to a file, a group of files, or a portion of a file, based on a standard mathematical algorithm applied to the characteristics of the data set. The most commonly used algorithms, known as MD5 and SHA, will generate numerical values so distinctive that the chance that

This mathematical property of hash makes hashing the ideal tool for authentication of electronic evidence.⁵² As will be explained further, this is one of hashing's key features and the primary reason the legal profession should abandon Bates for hash. Hash not only provides a unique identifying name for every computer document or other ESI, it also guarantees that electronic evidence has not been altered, either by accident or malicious intent.⁵³

Software to run both the SHA-1 and MD5 hash analysis of files is widely available, easy to use, and many are free.⁵⁴

D. Examples of Hash

The hash value of any file can be quickly calculated, regardless of the type of electronic file, including graphics. For instance, the hash values of the instant Word document are:

MD5: 588BCBD1845342C10D9BBD1C23294459
SHA-1: C24AE3125BFDBCE01A27FDDA21B3A7E83FAFF69E

If the author changes only the colon at the sentence above to a period, all else remaining the same, the hash values are now:

MD5: 5F0266C4C326B9A1EF9E39CB78C352DC
SHA-1: 4C37FC6257556E954E90755DEE5DB8CDA8D76710

Although the two files have only this trivial difference, there are no similarities in these hash values, illustrating that hashing will detect even the slightest file alteration.

any two data sets will have the same hash value, no matter how similar they appear, is less than one in one billion.

Id.

52. The *Pocket Guide for Judges'* explanation of hash supports this proposition: "Hashing" is used to guarantee the authenticity of an original data set and can be used as a digital equivalent of the Bates stamp used in paper document production." *Id.*

53. *Id.*

54. A HashTab Shell Extension to Windows is available for free as of May 29, 2007. See [Beeblebrox.org](http://www.beeblebrox.org), Hash Tab Shell Extension, available at <http://www.beeblebrox.org/software.php> (last visited May 29, 2007). Another free hash analysis software program intended for forensic examination is "PinpointHash." Pinpoint Labs, Pinpoint Hash, at http://www.pinpointlabs.com/free_tools/hash/ (last visited May 29, 2007).

E. The Irreversibility of Hash

Hash algorithms in cryptography are considered a type of encryption method; one that creates an irreversible encoding of a file such that the original message can never be deduced by the encrypted form of the message, the hash value.⁵⁵ Normally encryption algorithms are used to provide confidentiality to a message, and are reversible, in that the original message can be restored by applying the correct, secret decryption key.⁵⁶ In other words, the encryption process is necessarily a two-way, reversible process because the whole point is to have a secret, encrypted message that can still be read by the intended recipient.⁵⁷ But with hash encryption the process is irreversible. The encryption is one-way only. The original file cannot be restored from the hash value; even the basic properties of the original file remain hidden.⁵⁸

The irreversibility of hash is its most valuable quality to cryptographers. It makes it possible to verify the authenticity of a file while still maintaining the complete secrecy of the actual contents of a file. Thus, for instance, it can be used to safeguard the secrecy of passwords on a computer system because only the hash values of users' passwords need to be stored on the system database.⁵⁹

As will be later explained in more detail, the same irreversible quality of hash makes it ideal for use in digital signature processes. These hash-based processes guarantee the identity of the creator of a computer file, in the same way a handwritten signature authenticates the creator of a paper document.⁶⁰ The ability of hash to enable ironclad digital signature

55. FRED PIPER & SEAN MURPHY, CRYPTOGRAPHY: A VERY SHORT INTRODUCTION 70-71 (2002); Gregory L. Fordham, E-Discovery: Get Ready to Apply the New FRCP Changes 11, 109 (Course from National Business Institute 2006); GOVE ET AL., *supra* note 50, ch. 4, § 32.

56. PIPER & MURPHY, *supra* note 55, at 7-8, 15-17, 71-74. An "encryption key," which may be either public or private, is applied to an encryption algorithm to create a disguised, encrypted version of a message. The encrypted message, called a "cryptogram" or "ciphertext," is then restored to its original intelligible "plaintext" version by applying a secret "decryption key" using the same encryption algorithm. The original message is thereby restored. The encryption process is thus reversible, and indeed, that is the whole point so as to facilitate secret communications that cannot be read by anyone intercepting the message who does not know the encryption algorithm and decryption key.

57. GOVE ET AL., *supra* note 50, ch. 4, §§ 1, 2, 32.

58. *See, e.g.*, FastSum Software, *supra* note 43.

59. PIPER & MURPHY, *supra* note 55, at 70.

60. *Id.* at 93-99; *see also* Biddle, *infra* note 124 (describing how that works).

programs is a key application of hash; indeed it is the original reason MIT Professor Rivest invented the MD5 hash.⁶²

F. *The Value of Irreversibility in e-Discovery*

The irreversibility of hashing makes it possible to perform a hash search of a computer for specific hash values without revealing the actual contents of the computer searched. The search can only reveal whether the identical files are present. This is explained in *Creative Science Systems, Inc. v. Forex Capital Markets, LLC*, a trade secret theft case:⁶³

EvidentData shall limit its inspection of Defendant's computer network to copying the configuration file of any load balancing servers and running a utility program that generates a specific digital signature (a "MD5 hash value") for each file on the FXCM computer servers to generate a file listing along with each file's corresponding MD5 hash value. EvidentData will retain these results in a computer text file. EvidentData shall be permitted to compare the MD5 hash values for the files on the FXCM servers with the MD5 hash values for files unique to the NetZyme software. If EvidentData identifies files unique to the NetZyme software, EvidentData shall make a forensic image or logical backup of the file, at the discretion of the EvidentData computer forensic examiner on site.⁶⁴

The irreversibility of hash makes it well suited for electronic discovery in situations involving the search of confidential ESI. This is a common scenario for intellectual property theft cases such as that in *Creative Science Systems*.⁶⁵ Since a hash analysis cannot reveal the contents of any previously unknown file, a hash search of another's computer will not compromise its security. Unless there is a match in hash values to a known file, the contents are not deducible.⁶⁶ Conversely, if the computer searched has stolen data or software, then, as shown in *Creative Science Systems*, it can be detected from its known hash values. This kind

61. GOVE ET AL., *supra* note 50, ch. 4, § 33.

62. Rivest, *supra* note 37, Executive Summary, at 1 n.23.

63. 2006 WL 870970, at *4 (N.D. Cal. 2006).

64. *Id.* at *4.

65. Fordham, *supra* note 55, at 111.

66. *Id.* at 109; PIPER & MURPHY, *supra* note 55.

of hash comparison will provide conclusive evidence of the existence, or not, of identical files.⁶⁷

Hash is an excellent tool to search ESI because it is fast, and it cannot be easily misled. Hash is hard to fool because hashing only analyzes a file's contents to derive the unique hash value of the file. It may seem odd, but the name of a file is not part of the contents of the file.⁶⁸ Instead a file name is stored outside of the file itself as part of the operating system filing indexes.⁶⁹ For that reason the hashing process does not examine or include a file name.

This means that a computer file cannot be hidden from hash detection by changing its name and placement, a common practice by "evil doers" attempting to hide ESI in a computer.⁷⁰ Instead, only a change in file contents can change the hash value, and thus elude a hash search. For instance, you can easily hide a stolen file from word searches, and from human detection, simply by changing its name, extension, and location. For example, you could change a Word file with a .DOC extension, to an .EXE extension (change "smoking-gun.doc" to "innocuous.exe") and move the file to a Windows system directory that normally has many other executable (.EXE) program files. With this false extension and location the stolen Word file would be camouflaged and hidden from other inspection. But since a hash search does not include these parameters, this disguise will not impact a hash analysis at all.⁷¹ The same applies to an attempt to hide a known virus by name change and innocuous placement.⁷²

An individual can also use this property of hash to search some files that have been deleted from a computer, since a deleted file still remains on the hard drive (until it is written over), and only its name and references have been deleted. *United States v. Eberle*⁷³ contains a good description of a forensic examination of a reformatted "wiped" hard drive, showing how "deleted files" can still be detected and identified by hash, or the opposite

67. In a recent Wisconsin state court case, the search of a former employee's computer did not uncover any matching hash value files, and so the court refused to allow further forensic analysis to determine if the employee had reformatted the computer, characterizing that as a "fishing expedition." *Liturgical Publ'n, Inc. v. Karides*, 2006 Wis. App. LEXIS 313, at 18 (Wis. Ct. App. 2006). The court also noted that "hash values are defined by the parties as alphanumeric identifiers of files." *Id.* n.7.

68. Fordham, *supra* note 55, at 111-12.

69. *Id.*

70. *Id.*

71. *Id.*

72. Method and System for Limiting Processor Utilization by a Virus Scanner, U.S. Patent No. 7,085,934 (filed July 27, 2000) (issued Aug. 1, 2006) [hereinafter Patent No. 7,085,934].

73. No. CRIM 05-26 ERIE, 2006 WL 1705143 (W.D. Pa. 2006).

can be proven, that those files were not there. In *Eberle* the hash search showed the ESI at issue were not on the computer:

Detective Lynn then performed a more targeted search known as a “hash value check,” whereby she searched for a specific identifier, known as an MD5 hash, that is particular to an internet image, much like a fingerprint. This hash check similarly failed to uncover any of the images that had been uploaded onto the Yahoo! Server in 2001.⁷⁴

V. THE APPLICATION OF HASH TO AUTHENTICATE ESI

There are many advantages for litigation management to using hash values instead of Bates stamps. Prime among them is the ability of hashing to disclose any difference in computer files. As demonstrated previously by the hash of this Article, the hash values for two files will be completely different if the file contents are not one hundred percent identical. This is how using hash confirms with complete certainty whether a file has been altered. Hash is for this reason an excellent tool to guarantee the authenticity of ESI,⁷⁵ and as we will see, it has been accepted as reliable for that purpose by courts throughout the country.⁷⁶

Id. at *2.

74. The well-known electronic discovery scholar, Judge Paul Grimm, seems to agree. *Lorraine v. Markel Am. Ins. Co.*, PWG-06-1893, at 25-26 (D. Md. May 4, 2007). In this 101 page decision by Judge Grimm, he sets forth detailed evidentiary guidelines on the admissibility of ESI. Judge Grimm explains that hash provides a method of authenticating electronic evidence under Rule 901(b)(4):

Hash values can be inserted into original electronic documents when they are created to provide them with distinctive characteristics that will permit their authentication under Rule 901(b)(4). Also, they can be used during discovery of electronic records to create a form of electronic “Bates stamp” that will help establish the document as electronic.

Id.

76. See POCKET GUIDE FOR JUDGES, *supra* note 51. No case was uncovered where hash was rejected as a means of ESI authentication.

A. The Special Importance of Hash in Native File Productions

Although production of ESI in native format was not typical until recently, that has begun to change dramatically under the new Federal Rules of Civil Procedure (FRCP) which went into effect on December 1, 2006. The new FRCP encourage the production of ESI in native format, in fact many contend it is now the default mode of production.⁷⁸

The increased use of the native file format makes the authenticity powers of hash all the more important because ESI in native format is easy to modify, both intentionally and accidentally.⁷⁹ Without hash, many slight

77. Sedona Conference Glossary, *supra* note 38, at 29. “Native Format” is defined as follows:

Electronic documents have an associated file structure defined by the original creating application. This file structure is referred to as the “native format” of the document. Because viewing or searching documents in the native format may require the original application (for example, viewing a Microsoft Word document may require the Microsoft Word application), documents are often converted to a vendor-neutral format as part of the record acquisition or archive process. “Static” formats (often called “imaged formats”), such as TIFF or PDF, are designed to retain an image of the document as it would look viewed in the original creating application but do not allow metadata to be viewed or the document information to be manipulated.

Id.

78. FED. R. CIV. P. 34(b). The new rules do not use the term “native format.” Instead, the rules require production of the electronically stored information (ESI) “as they are kept in the usual course of business or . . . in a form or forms . . . that are reasonably usable.” *Id.* The usual course of business is to keep ESI in its native format because that is how it is used. One reason for the alternative of production in a reasonably usable form is to prevent a party from using native format as a kind of non-production because the opposing party may not have the ability to read the ESI in its native format. The opposing party may not have access to the application in which the ESI was created. The parties can, of course, still request other non-native formats, such as TIFF. They can also object to a native format production request and argue that another format such as TIFF is more reasonably usable. Still, native format production now seems to be the default mode under the new rules, and native file production is likely to become prevalent in the coming years. *See, e.g., Williams v. Sprint/United Mgmt. Co.*, 230 F.R.D. 640, 656 (D. Kan. 2005) (Williams I); Withers, *supra* note 22, at 188. At present, however, most e-discovery production is made in either TIFF or PDF format. Many e-discovery vendors who have invested heavily in software designed to create and search TIFF oppose the trend towards native format file production and PDF formats. Some courts also oppose it as an unnecessary expense. *Wyeth v. Impax Labs*, No. 1 :06-cv-222-JJF, 2006 U.S. Dist. LEXIS 79761, at *4 (D. Del. Oct. 26, 2006); *see Kentucky Speedway, LLC v. NASCAR*, 2006 U.S. Dist. LEXIS 92028 (E.D. Ky. Dec. 18, 2006).

79. As an example of an accidental change to a file by inadvertent modification of a file’s internal metadata, if you just open a Word file, and do not do anything other than save it again, with the same name, and with no changes or other activity whatsoever, you will still be changing the file and a new hash will result. It has been changed by the mere act of resaving the file because

but possibly significant changes would be hard to detect in large files, and harder still to prove.

Native files are, in the author's opinion, likely to become the preferred format for most (but not all) ESI production under the new rules, primarily because the native format guarantees disclosure of all of the information in an ESI file, including its internal metadata.⁸⁰ One of the compelling reasons for the recent changes to the FRCP is to require this full disclosure.⁸¹ The changes discourage the conversion by the producing party of native files into another format that hides data or degrades searchability.⁸² This is exactly what happens when, for instance, a party produces Excel spreadsheets in a TIFF or PDF or a paper format. This conversion from the native format removes all of the formulas embedded in the Excel files and thus greatly reduces the usability of the spreadsheets. The same can be true for the conversion of many other applications, where metadata and search features are easily lost if particular precautions are not

the last saved date of a Word file is one of its properties and is maintained as part of the file's own internal metadata. Conversely, if you open the file, and just close it without saving it, the internal metadata remains unchanged and so the hash value of the file remains the same. Note that in the second example the external system metadata on that file will change by just opening and closing it, since the system monitors "last accessed" date. External metadata does not impact the file hash because it is not part of the file. The author knows this from experimentation with hash values.

80. See lengthy discussion about metadata production and native files at Ralph C. Losey, e-Discovery Team Blog, available at <http://ralphlosey.wordpress.com/meta-prod/> (last visited May 24, 2007).

81. See Withers, *supra* note 22, at 188. Withers, who is the Director of Judicial Education and Content for "The Sedona Conference," explains the background:

The files in native formats are dynamic, and behave the way they do in the active business environment, which may be significant to understanding their function and content. They also contain non-apparent information, such as metadata (embedded records of the creation and management of the document), editorial comments and changes (which may be kept in the native file format for later revision), and functions (such as the mathematical formulas that determine the relationship of cells in a spreadsheet or records in a database). The form of production is more than a question of convenience or cost—it becomes a question of relevance and "best evidence," as it applies to electronically stored information.

Id.

82. Committee Notes to the 2006 Amendments to Rule 34 state: "If the responding party ordinarily maintains the information it is producing in a way that makes it searchable by electronic means, the information should not be produced in a form that removes or significantly degrades this feature." FED. R. CIV. P. 34, Advisory Committee Notes 2006 Amendment Subdivision (b).

taken.⁸³ Of course, it is sometimes necessary for attorneys to intentionally alter documents and ESI by redacting them to protect the disclosure of privileged communications, but in those situations the rules require the creation of a privilege log to disclose and justify the action, and the maintenance of the original unaltered document or ESI.⁸⁴

Native files reveal all information, including all of their internal metadata, and are easy to search, but, as mentioned, they are also easily altered. For instance, a native Word file can be opened and the date on the face of the document easily changed.⁸⁵ Since dates in litigation (who knew what when) are frequently critical, this one small change to a document could have a major impact on a case.⁸⁶ Since the new rules emphasize native file productions, the authentication properties of hash have become more important than ever.⁸⁷

B. Hash is Widely Accepted in Civil Cases

Even before the implementation of the new rules, most e-discovery commentators recommended the adoption of hash for authenticity purposes. For instance, Michael Arkfeld, in his seminal text on e-discovery, states that “[i]n order to prevent any allegation that produced electronic data has not been altered it is suggested that a hash value be generated for electronic discovery computer files.”⁸⁸ The U.S. District Court for the District of Maryland has even included a suggestion that the parties use hash and hash marks in its local rules.⁸⁹

83. See, e.g., *In re Verisign, Inc.*, No. C 02-02270 JW, 2004 WL 2445243 (N.D. Cal. Mar. 10, 2004). In this class action securities suit, the defendants were ordered by the magistrate judge to produce all documents in native electronic form. *Id.* at *3.

84. FED. R. CIV. P. 26(b)(5). At the present time it is difficult to redact some types of native files. For that reason, native files are often converted to TIFF format, and then redacted. Adam I. Cohen & David J. Lender, *Electronic Discovery: Law and Practice* §§ 9.05[B], 9.06[A] (Supp. 2007).

85. See *Plasse v. Tyco Elec. Corp.*, 448 F. Supp. 2d 302 (D. Mass. 2006). The plaintiff in *Plasse* tried to alter the date on his resume, but the defendant’s computer forensics expert exposed his attempts and, as a result, the plaintiff’s case was dismissed. *Id.* at 306, 311.

86. See, e.g., *Zubulake v. UBS*, 229 F.R.D. 422 (S.D.N.Y. 2004). In this important e-discovery case, determining when Zubulake’s supervisor was aware of an EEOC charge was critical to proving her retaliation claim. *Id.* at 430. Her supervisor testified at deposition that he did not know about her charges when he fired her. *Id.* But an e-mail he thought he had destroyed was later found, and this proved he did know and had lied. *Id.*

87. Conversely, metadata and hash are of no importance to paper discovery, where the terms do not even apply.

88. MICHAEL R. ARKFELD, *ELECTRONIC DISCOVERY AND EVIDENCE* § 5.5(G) (2006).

89. U.S. District Court for the District of Maryland, *Suggested Protocol for Discovery of*

The use of hash to authenticate ESI is discussed in detail in a landmark case in e-discovery, *Williams v. Sprint/United Management C. (Williams I)*⁹⁰. *Williams I* is an age discrimination class action lawsuit where the defendant, Sprint, initially produced Excel spreadsheets in TIFF format.⁹¹ Plaintiffs complained, stating they needed native files so they could see the formulas and analyze the files.⁹² Sprint delayed, but finally produced the native Excel files. Plaintiffs then learned that the Excel files had all been scrubbed of metadata and certain cells were locked so they could not be accessed.⁹³

The court had ordered Sprint to produce the Excel records as native files “in the manner in which they were maintained,” in other words, unaltered. For this reason, upon complaint by plaintiff, the court ordered Sprint to show cause why sanctions should not be entered for its unauthorized metadata scrubbing.⁹⁴ Sprint argued that there was an emerging standard against the production of metadata relying primarily on the highly regarded *Sedona Principles for Electronic Document Production*,⁹⁵ especially Principle 12, which states that “[u]nless it is material to resolving the dispute, there is no obligation to preserve and produce metadata absent agreement of the parties or order of the court.”⁹⁶ Sprint also argued that it had to lock the spreadsheet cells so they could not be altered, either by accident or intent.⁹⁷

The court discussed metadata at length, including what it is, why it can be important, and what the commentaries, primarily Sedona, and case

Electronically Stored Information, at 20, <http://www.mdd.uscourts.gov/news/news/ESIProtocol.pdf> (last visited May 24, 2007) (encouraging parties to discuss use of hash values or “hash marks” when producing electronic records in discovery to facilitate their authentication).

90. 230 F.R.D. 640 (D. Kan. 2005).

91. *Id.* at 642.

92. *Id.*

93. *Id.* at 645.

94. *Id.* at 644-45.

95. The Sedona Conference Institute and its publications on electronic discovery are highly regarded and frequently quoted by courts and commentators. Its key publication, which was discussed in *Williams I*, is *The Sedona Principles Addressing Electronic Document Production*.

THE SEDONA CONFERENCE, THE SEDONA PRINCIPLES ADDRESSING ELECTRONIC DOCUMENT PRODUCTION (July 2005 version) [hereinafter SEDONA PRINCIPLES], available at http://www.thesedonaconference.org/publications_html?grp=wgs110 (last visited (May 24, 2007).

96. The commentary to this principle opined that “most of the metadata has no evidentiary value, and any time (and money) spent reviewing it is a waste of resources.” The commentary also set forth an important exception to its principle 12: “Of course, if the producing party knows or should reasonably know that particular metadata is relevant to the dispute, it should be produced.” *Id.*

97. *Williams*, 230 F.R.D. at 655.

law suggest is the emerging trend as to metadata scrubbing or production.⁹⁸ The court accepted the Sedona Principle 12 as an important part of the “emerging standard,” but rejected Sprint’s argument that this meant the Excel spreadsheets’ metadata should not be produced.⁹⁹ Instead, the court found the Excel metadata was material to the dispute, and Sprint should have known and produced it.¹⁰⁰ The court held that metadata must be produced absent agreement or court order, and ordered the reproduction without metadata-scrubbing or cell locking.¹⁰¹

The court rejected Sprint’s authenticity argument on the basis of hash:

Defendant’s concerns regarding maintaining the integrity of the spreadsheet’s values and data could have been addressed by the less intrusive and more efficient use of “hash marks.” For example, Defendant could have run the data through a mathematical process to generate a shorter symbolic reference to the original file, called a “hash mark” or “hash value,” that is unique to that particular file. . . . This “digital fingerprint” akin to a

98. *Id.* at 646.

99. *Id.* at 650.

100. *Id.* at 652-54.

101. The court actually goes well beyond Sedona’s Principle 12 and establishes a default standard for native file production where the producing party must justify production in another format before it will be permitted:

Based on these emerging standards, the Court holds that when a party is ordered to produce electronic documents as they are maintained in the ordinary course of business, the producing party should produce the electronic documents with their metadata intact, unless that party timely objects to production of metadata, the parties agree that the metadata should not be produced, or the producing party requests a protective order. The initial burden with regard to the disclosure of the metadata would therefore be placed on the party to whom the request or order to produce is directed. The burden to object to the disclosure of metadata is appropriately placed on the party ordered to produce its electronic documents as they are ordinarily maintained because that party already has access to the metadata and is in the best position to determine whether producing it is objectionable. Placing the burden on the producing party is further supported by the fact that metadata is an inherent part of an electronic document, and its removal ordinarily requires an affirmative act by the producing party that alters the electronic document.

Williams I, 230 F.R.D. at 652. In a subsequent order, this procedure was applied to plaintiff’s request for the metadata in e-mail and Sprint’s objection was this time sustained because it proved undue burden. *Williams v. Sprint/United Mgmt. Co. (Williams II)*, 2006 WL 3691604 (D. Kan. Dec. 12, 2006).

tamper-evident seal on a software package would have shown if the electronic spreadsheets were altered. When an electronic file is sent with a hash mark, others can read it, but the file cannot be altered without a change also occurring in the hash mark. . . . The producing party can be certain that the file was not altered by running the creator's hash mark algorithm to verify that the original hash mark is generated. This method allows a large amount of data to be self-authenticating with a rather small hash mark, efficiently assuring that the original image has not been manipulated.¹⁰²

In a subsequent order in this case, *Williams v. Sprint/United Mgmt. Co. (Williams II)*,¹⁰³ this same analysis was followed to consider a later request by the plaintiff for the metadata in e-mail. This time Sprint's objection was sustained because Sprint proved it would be an undue burden to produce the e-mails in native format. The metadata request was denied in *Williams II* in part because the court found that the provision of hash values for all of the attachments associated with e-mails was adequate to allow the plaintiff to match up the attachments with the e-mails. For this reason, a second, expensive reproduction of e-mails in native format with all metadata was an unnecessary burden.¹⁰⁴

Hash not only protects litigants from unscrupulous or negligent adversaries or experts who might try to alter computer files, it also allows both producers and recipients of productions to prove the original was not altered.¹⁰⁵ Judge Grimm makes this point in *Lorraine v. Markel American Insurance Co.*,¹⁰⁶ an opinion providing a treatise on the admissibility of electronic evidence, where he notes:

A party that seeks to introduce its own electronic records may have just as much difficulty authenticating them as one that attempts to introduce the electronic records of an adversary. Because it is so common for multiple versions of electronic documents to exist, it sometimes is difficult to establish that the version that is offered into evidence is the "final" or legally operative version. This can plague a party seeking to introduce a favorable version of its own electronic records, when the adverse

102. *Williams I*, 230 F.R.D. at 655.

103. *Williams II*, 2006 WL 3691604.

104. *Id.* at *8.

105. Fordham, *supra* note 55, at 11.

106. *Lorraine v. Markel Am. Ins. Co.*, PWG-06-1893, at 26 (D. Md. May 4, 2007).

party objects that it is not the legally operative version, given the production in discovery of multiple versions. Use of hash values when creating the “final” or “legally operative” version of an electronic record can insert distinctive characteristics into it that allow its authentication under Rule 901(b)(4).¹⁰⁷

Since an entire drive can be hashed, not just one file or group of files, it is possible to conclusively prove that the files or hard drive a lawyer examines have not been changed.¹⁰⁸ You need only compare the hash numbers. If the hash of the original and copy are identical, that is conclusive proof nothing was altered.¹⁰⁹ Hashing, unlike Bates numbers, allows a lawyer to protect an ESI file from alteration and guarantees its authenticity.

C. Hash is also Widely Used in Criminal Cases

The authenticity guarantees provided by hash have made it an indispensable tool in criminal investigations involving the misuse of

107. *Id.*

108. Fordham, *supra* note 55. Fordham recommends verification that a hard drive has been correctly copied, also called “imaged,” in one of two ways, both of which involve hashing:

Even when tested and reliable imaging tools have been selected for making the image, current best practices require that the image be verified by at least one of two options. The first is by re-imaging with different equipment and then comparing the MD5 hash, or equivalent, of the first image to the MD5 hash of the second. In the alternative, one could also compare the MD5 hash or equivalent of the original drive to the MD5 hash of the image. Either method will require at least two passes over the original drive.

Id.

109. WILLIAM L. NORTON, JR., NORTON BANKRUPTCY LAW AND PRACTICE § 141.40 (2d ed. 2007). Norton recommends making multiple forensic images of the debtors’ hard drives and then hashing the original and images to verify authenticity. But see “Bit Flipping,” which, among other things, is slang in computer forensics for a spontaneous switch or “flip” in the value of a bit from 0 to 1, or from 1 to 0. This change in a bit’s value is a naturally occurring type of data corruption that is inherent in today’s technology, primarily in hard drives. It can have many causes, including spontaneous, very slight, variations over time of the magnetic fields that hard drives use to store bits of information. Due to this decay, a bit value can change from 0 to 1, or 1 to 0. Either way this minor, usually otherwise undetectable, corruption of data will create a new hash value for the hard drive (or file). This phenomena can be taken into account and corrected by bit mapping software that allows you to track and identify any such “bit flips” on the hard drive. You can then restore the original 0 or 1 value of the degraded bit on the hard drive and rerun the hash. This information is based on a conversation with a national forensic expert, Benjamin R. Cotton, Director of Forensics for Emerging Technologies Group in Herndon, Virginia; and personal experience.

computers. For instance, most online pornography cases depend upon hashing to locate and then prove at trial the presence of child pornography on a defendant's computers.¹¹⁰ In fact, computer forensics experts employed by law enforcement today spend a large part of their time searching personal computers and web servers for child pornography.¹¹¹ A recent child pornography case, *United States v. Cartier*,¹¹² shows how hash and the latest peer-to-peer (P2P) Internet technology (which will be explained in the next section) also depends on hash. The ability to search P2P networks for matching hash files allows police anywhere in the world to locate and find child pornography on the computers of individuals who use these networks. Here the Computer Crime Unit of Spain (Spanish Guardia) advised the FBI that by using a P2P search it had located a man in North Dakota with substantial amounts of child pornography on his computer. The P2P search is described in the opinion as follows:

The Spanish Guardia was using a software program it developed with a private company to search the Edonkey peer-to-peer ("P2P") computer network to search for people who possessed child pornography. The software allowed the Spanish Guardia to search for child pornography by using hash values. A hash value is a unique multi-character number that is associated

110. See Richard P. Salgado, *Fourth Amendment Search and the Power of the Hash*, 119 HARV. L. REV. F. 38 (2006) (providing a good technical description of hash and its search and seizure implications). Salgado's article begins with this description of hash:

Hashing is a powerful and pervasive technique used in nearly every examination of seized digital media. The concept behind hashing is quite elegant: take a large amount of data, such as a file or all the bits on a hard drive, and use a complex mathematical algorithm to generate a relatively compact numerical identifier (the hash value) unique to that data. Examiners use hash values throughout the forensics process, from acquiring the data, through analysis, and even into legal proceedings. Hash algorithms are used to confirm that when a copy of data is made, the original is unaltered and the copy is identical, bit-for-bit. That is, hashing is employed to confirm that data analysis does not alter the evidence itself. Examiners also use hash values to weed out files that are of no interest in the investigation, such as operating system files, and to identify files of particular interest. It is clear that hashing has become an important fixture in forensic examinations.

Id. ¶ 1.

111. Douglas Rehman, *Electronic Discovery: Everything you Always Wanted to Know Before It's Too Late*, Computer Forensics; Presentation at Florida Bar Continuing Legal Education Seminar 0286R (Jan. 20, 2006).

112. 2007 WL 319648 (D. N.D. Jan. 2007).

with a computer file. Some computer scientists compare a hash value to an electronic fingerprint in that each file has a unique hash value. The Spanish Guardia was using an investigative strategy in which it would start with a known image of child pornography, take the hash value associated with that file, enter the hash value in its P2P software to search for others possessing the same image of child pornography, and record the internet protocol (“IP”) address associated with the other person’s computer if that person shared more than five known images of child pornography. In conducting its search, the Spanish Guardia did not open the file it located to confirm it was child pornography.¹¹³

The district court in North Dakota upheld the search warrant subsequently issued to the FBI, and found probable cause from the P2P search, even though the contents of the hashed files were never actually viewed to confirm a match before the search warrant was issued. The court held that:

While relying on a hash value alone would doubtfully meet a certainty standard, Judge Senechal only had to find probable cause. While the use of hash values is not full proof, few things are. Agent Boeckers possessed information from a reliable law-enforcement agency, the Spanish Guardia. The Spanish Guardia relied on a trustworthy means of computer forensics. Therefore, this reliable information established probable cause to issue a warrant.¹¹⁴

D. Commercial and Governmental Uses of Hash

Many state and local governments are abandoning paper ballots in favor of computerized voting. Hash is used in computerized voting to guard against software tampering. The hash values are verified at various times to guarantee that the voting machine software has not been altered from its original installation.¹¹⁵ Hash values have also been used by the

113. *Id.* at *1.

114. *Id.* at *3.

115. See, e.g., Celeste Biever, *U.S. Boosts e- Voting Software Security*, [NEWSCIENTIST.COM NEWS SERV.](http://www.HERALDTribune.com/apps/pbcs.dll/article?AID=/20061128/NEWS/61128001-1/NEWS0521) (Oct. 28, 2004), at <http://www.HERALDTribune.com/apps/pbcs.dll/article?AID=/20061128/NEWS/61128001-1/NEWS0521> (last visited May 24, 2007); Mares & Company, *Data Integrity: How to Authenticate Your Electronic Records, How to Use Maresware to Validate*,

government to enforce laws prohibiting the sending of sexually explicit materials to hashed e-mail addresses that have been registered with the state as belonging to minors.¹¹⁶

There are many commercial applications of hash in a variety of fields. For example, banks and other institutions routinely use hash to verify that their software or databases have not been altered, and for other authentication purposes, especially involving online banking.¹¹⁸

Hash is also used as a basis for computer virus scanning where the hashes of known viruses are scanned on potentially infected computers. This has been the basis of several recent patents.¹¹⁹ Hash has also been used in various computer system security devices, where again several patents have issued for inventions that utilize the authentication properties of hash.¹²⁰ In fact, a Google patent search of “hash values” uncovers 547 patents that mention “hash value” as a part of the described invention in a wide variety of fields and applications.¹²¹

At least one commercial application of hash designed to prevent the unauthorized use of certain parts and machinery has led to litigation.¹²² A printer manufacturer used the authentication properties of hash to prevent any off-brand toner cartridges from functioning on its printers.

E. Electronic Data Transfers

Another common use of hash today by individuals, businesses, and governments is to verify that a data transmission or file download has been

Voting Machine Software, at http://www.dmares.com/maresware/articles/hash_faqs.htm#VOTING (last visited May 24, 2007).

116. Free Speech Coalition, Inc. v. Shurtleff, 2007 WL 922247, at *2 (D. Utah, Mar. 23, 2007) (“The only way that emailers can determine whether or not any particular email address in Utah has been registered under the CPR is to register to participate in the Registry’s so-called ‘scrubbing’ services and have Unspam compare hash values to look for matches with any contact point on the Registry.”).

117. GOVE ET AL., *supra* note 50, ch. 4, §§ 1, 2, 32.

118. See, e.g., Berry Schoenmakers, Basic Security of the ECASH(TM) Payment System, Technical Univ. of Vienna, Austria, at <http://www.econ.tuwien.ac.at/lva/elgeld/ps/literatur/basic%20security%20of%20the%20ecash%20payment%20system.pdf> (last visited May 24, 2007); Check Based Online Payment and Verification System and Method, U.S. Patent No. 7,069,250 (filed Oct. 15, 2001) (issued June 27, 2006).

119. See, e.g., Patent No. 7,085,934, *supra* note 72.

120. See, e.g., Secure Printing Method, U.S. Patent No. 6,711,677 (filed July 12, 1999) (issued Mar. 23, 2004).

121. Hash Value, Google Search, <http://www.google.com/patents?q=%22hash+value%22&btnG=Search+Patents> (including a search of all patents up to the middle of 2006). To search patents, see <http://www.google.com/patents> (last visited Apr. 15, 2007).

122. Lexmark Intern, Inc. v. Static Control Components, Inc., 387 F.3d 522 (6th Cir. 2004).

received without corruption or data loss.¹²³ Hash can also verify a digital signature of the sender of the transmission, and thus authenticate the identity of the sender.¹²⁴ Digital signatures with hash authentication have many applications today, including time confirmation for use with patents and digital art authentication services.¹²⁵ One such “digital notary” company, Surety Technologies, even publishes the hash values of electronic documents it registers in the *New York Times*. This provides easy proof of the content and date of authentication.¹²⁶

123. See, e.g., [LabCompliance.com](http://www.labcompliance.com) Information Page, Verification of File Integrity with MD5 Hash Calculations, at <http://www.labcompliance.com/info/2003/06/030616-md5.htm> (last visited May 24, 2007).

124. See C. Bradford Biddle, *Misplaced Priorities: The Utah Digital Signature Act and Liability in a Public Key Infrastructure*, 33 SAN DIEGO L. REV. 1143, 1149-50 (1996).

Digital signatures, as contemplated under the Utah Act, involve another step: the one-way hash function. . . . If Alice wants to “sign” an electronic document with a digital signature and send it to Bob, she does not have to encrypt the entire document with her private key. Instead, she can run the document through a one-way hash function, creating a message digest. She can then encrypt that message digest using her private key and send it along with the unencrypted document. Note that every digital signature is unique to the document for which it is created. So a forger could not take Alice’s digital signature from one document, append it to a fraudulent document, and then successfully claim that Alice had signed the fraudulent document. When Bob receives the message, he independently runs the same one-way hash function on the original message to determine what the message digest should be. He then decrypts (or “verifies”) Alice’s digital signature, using Alice’s public key. If the message digest in Alice’s decrypted digital signature matches the message digest that Bob calculated from the message on his own, then Bob knows that the message is indeed from Alice, and that it has not been altered since she signed it. If the message digests are not identical, then Bob knows that Alice did not sign the same message that he received—somehow the message has been altered. If the message digests are identical, Alice cannot later successfully claim that she did not send the message. No one else could have created the digital signature attached to the document. Thus Alice and Bob may have achieved the qualities of data origin authentication, message integrity, and non-repudiation.

Id.; Dean M. Harts, *Reel To Real: Should You Believe What You See?*, 66 DEF. COUNS. J. 514, 522-23 (1999) (article on the authentication of visual images also provides a good description of the use of hash in digital signature verification).

125. ALEXANDER LINDEY & MICHAEL LANDAU, *LINDEY ON ENTERTAINMENT, PUBLISHING AND THE ARTS* § 19.2111 (3d ed. 2007); Dean M. Harts, *Reel to Real: Should You Believe What you See?*, 66 DEF. COUNS. J. 514, 522-23 (1999) (providing a good description of the use of hash in digital signature verification).

126. See *supra* note 125.

A recent district court case in Florida considered a patent dispute involving competing digital signature and time stamping.¹²⁷ The competing software inventions both use hash to validate when a document is created and by whom. The opinion describes the hash functioning of the challenged invention in some detail, showing again how digital signatures and authentication of electronic documents depend on hashing.¹²⁸

F. Federal Court Filings

A similar hash-based authentication system is used by all district courts in the United States. The federal system's mandatory Electronic Case Filing (ECF) program uses MD5 hashing to verify the digital signatures of attorneys who e-file pleadings, and to hash the pleadings and other documents that are e-filed with the court. Further, although most attorneys do not realize it, they receive a hash based confirmation every time there is an e-filing in one of their cases. The district courts use MD5 hashing to create and assign a unique identifying alphanumeric mark to all e-filings. In a matter of minutes¹²⁹ after any e-filing, all attorneys of record are e-mailed a "Notice of Electronic Filing." That e-mail specifically identifies the ESI by the name given to it by the filing party and by a very long computer-generated number called an "Electronic Document Stamp." The number is actually a 128-place alphanumeric. It includes the 32-place MD5 hash value of the pleading or other document e-filed, and the hash value of the filing attorney's digital signature.¹³⁰

127. *Timecertain, LLC v. Authentidate Holding Corp.*, 2006 WL 3804830, at *1 (M.D. FL. Dec. 22, 2006). The contested patent described hashing and its one-way quality as follows:

"Hashing" subjects a file's digital contents to an algorithm that effectively chops and mixes (i.e., "hashes") those contents to create a unique string of characters, which string is called a "digest" and which serves as a sort of digital "fingerprint" for that file. Hashing the same file with the same hashing algorithm produces the same digest. Hashing a different file produces a unique and different digest. Although a digest identifies a specific file, a digest is not the file and cannot be used to access or retrieve the file.

Id.

128. *Id.* at *2 (describing how hashing applies to the patented device).

129. Many attorneys with multiple cases in the same district court do not want to be constantly interrupted throughout the day with these e-mail notices. For that reason ECF allows for an alternate end-of-the-day notification wherein one e-mail provides notice of all activity in all cases in a particular district court.

130. Based on conversations with Dick Corelli, Sr., Public Affairs Specialist of the Administrative Offices of the U.S. Courts on Nov. 28, 2006, and with Bruce Walters, a software engineer and co-founder of Tyberia Development Group, Inc., on Nov. 17, 2006.

G. Peer-to-Peer Transfers

Perhaps the most frequent and public use of hash today arises from the relatively new P2P system of computer file distribution.¹³¹ This is a decentralized system whereby individual users can both upload and download files to each other on their computers via the Internet. It is most commonly used for direct “sharing” of media files, such as movies, videos, and MP3 songs, but is also widely used in online gaming. P2P allows users to locate and download music, movies, and other files on the computers of other users. The desired files are located and authenticated by search for published hash values of the media.¹³² Sometimes the web sites promoting P2P file transfers violate the copyrights of the media owners, and there has been frequent enforcement in this area, most notably in the highly publicized case of *A&M Records, Inc. v. Napster, Inc.*¹³³

A district court in Michigan recently considered a new case of this type, *Columbia Pictures Industries, Inc. v. Fysh*.¹³⁴ In *Fysh*, the court

131. See, e.g., Clay Shirky, What is P2P . . . And What Isn't (Nov. 24, 2000), at <http://www.openp2p.com/pub/a/p2p/2000/11/24/shirky1-whatisp2p.html> (last visited May 24, 2007).

132. The media files are often quite large, and so the files are sent in parts or packets, instead of one large file. This has led to the hashing of various parts of a media file, and to the creation of tables of the hash values of media, and sub-parts of the media, known as “hash lists” and “hash trees.” The entries in Wikipedia on these subjects are currently helpful to understand these and other new hashing applications behind P2P. See, e.g., Wikipedia, *Hash Tree*, http://en.wikipedia.org/wiki/Hash_tree (describing the uses of hash trees) (as of May 24, 2007, 10:00 EST).

133. 239 F.3d 1004 (9th Cir. 2001). The file sharing at Napster.com was held to constitute illegal copyright infringement, and this led to the closure of one of the most popular sites on the Internet. The full text of the opinion is available online at <http://www.riaa.com/news/filings/pdf/napster/napsterdecision.pdf>.

134. 2007 WL 541988 (W.D. Mich. Feb. 16, 2007). The copyright infringement suit was brought by Columbia Pictures, Disney Enterprises, and Warner Bros. Entertainment against Ben Fysh, a resident of the United Kingdom, who operated www.ed2k-it.com, which was hosted by Liquid Web, an Internet service provider located in Michigan. Since the web site was operated anonymously the original complaint was against unknown “John Does.” Expedited discovery led to the disclosure of the operator’s identity, Ben Fysh, who never appeared to defend the suit. The amended complaint alleges that Fysh “operated an eDonkey hash link site.” *Columbia Pictures Indus., Inc. v. Fysh*, USDC W.D. Mich., Civil Docket Case No. 5:06-CV-00037-RAE, Doc. #9 (Amended Complaint), ¶ 22. The plaintiff’s Motion for Entry of Default Judgment explains that:

An e-Donkey hash link site is a website that contains an index of files available on the eDonkey network (generally an extensive listing of movies and television programs, among other copyrighted content). The hash link site hosts and distributes small files known as “hashes.” Hashes do not themselves hold actual copies of a movie or television program. Rather, hashes are unique identifiers

entered a default judgment against a British web site operator who was not physically present in Michigan. The court found that it had personal jurisdiction under Michigan's long arm statute because the web site server computer was physically located in Michigan. A default judgment was entered for copyright infringement based on vicarious liability because the web site aided and encouraged others to make unauthorized copies of copyrighted movies and television programs. The web site did not actually contain the illegal copies, but it contained hash values of the illegal copies with links to other personal web sites where the materials could be downloaded in P2P fashion. The hash values allowed the web site users to find and identify media to upload and download to other users. This function of the web site was enough to sustain the default judgment and personal jurisdiction through actions in the forum state, Michigan. Hash played a key role in this finding as the court explained:

In the present case, Defendant operated and profited from a website that was "interactive." Defendant's website required users to download indexed hash files which corresponded to copyrighted movies or television programs. Defendant's website also allowed users to acquire login names to accomplish this. . . . This evidences Defendant's willful infringement because although

corresponding to particular files available on the eDonkey network—often a file containing a copyrighted movie or television program. Hashes automatically and invisibly instruct the eDonkey client program on a user's computer how and where to get the desired file. An eDonkey server manages the actual distribution of files, connecting uploaders (those who are distributing a movie) with downloaders (those who are copying a movie). This server functions in many respects like a "traffic cop," directing an eDonkey user's computer where to find users ("peers") who have a particular file, and then providing the user's computer with access to those other users to facilitate the download process. Hash link sites play an integral role in the process of using the eDonkey network to download files. Hash link sites both encourage users to upload hashes that uniquely correspond to copyrighted content and to index those hashes for easy retrieval by other users. Further, hash link sites perform a critical "quality control" function that allows users to efficiently download the best copies of movies and television shows on the eDonkey network. Hash link sites such as Defendant's are designed in part to weed out "bad" hashes and index only hashes linking to quality files. In the case of indexing television and movie hashes, hash link sites are intentionally designed to enable users to find the best copies of the unauthorized copyrighted works.

Id. Doc. #15, Statement of Facts.

within his control to delete the hash links and disable the infringement, he chose not to do so.¹³⁵

Fysh thus appears to be the first case in history where personal jurisdiction was based on the control and operation of hash. In effect, the operation of a mathematical algorithm, and posting of its results, was found to provide sufficient contacts with a state to satisfy the due process requirements of the Constitution. Both a money judgment in the amount of \$160,000 and a permanent injunction were entered.¹³⁶ This remarkable result shows the importance of hash algorithms to modern culture today, especially in the area of copyright protection.

VI. THE APPLICATION OF HASH TO FILTER ESI

A. De-Duplication

Filtration is the computerized process of reducing the total universe of possibly relevant ESI prior to review and production.¹³⁷ Filtration has become a standard ESI processing protocol in e-discovery. Filtration uses computerized processes to eliminate unresponsive files so that less time will be spent by attorneys to review irrelevant files. This culling process is, in my opinion, a key step to control e-discovery costs.

There are several automated methods to filter out irrelevant files. Perhaps the best known is the search method where software is utilized to search computers for files that contain certain words, word patterns, or concept patterns and employ boolean logic and even artificial intelligence.¹³⁸ The ESI containing matching search terms, or hits, are

135. *Id.* at *2-3.

136. Despite this injunction this web site is still operational at <http://www.ed2k-it.com> as of May 16, 2007. It boasts having 60,200 members, with an average of 35 new members every day.

137. The Electronic Discovery Reference Model Project has developed a nine-step model process for e-discovery projects that is now widely accepted by e-discovery vendors. See Electronic Discovery Reference Model (EDRM) Project Web Site, <http://www.edrm.net> (last visited May 24, 2007) [hereinafter EDRM Project Web Site]. The fifth step is “processing” of ESI. It is here that filtration of duplicate or known irrelevant files is accomplished through hash comparison. EDRM Processing Stages, Deduplication, at http://www.edrm.net/wiki/index.php/Processing_-_Processing_Stages#Deduplication (last visited May 24, 2007).

138. SEDONA PRINCIPLES, *supra* note 95, princ. 11, at 44 (stating that the “selective use of key ‘concept’ or word searches is a reasonable approach when dealing with large amounts of electronic data.”). The Sedona Conference Glossary, *supra* note 38 (defining “search” and related terms).

included in a responsive data set, and all other files are filtered out.¹³⁹ These same search methods are used in computerized legal research such as Lexis® and Westlaw® and as such are well known to most lawyers. Similar methods are employed by Internet search engines such as Google® that are also now commonly used by many lawyers to search for relevant information. In litigation today, the parties frequently understand and use search parameters to screen out irrelevant information, such as spam in e-mail and other obviously irrelevant materials.¹⁴⁰

De-duplication¹⁴¹ is a lesser-known filtration process, but is very important and should precede search culling.¹⁴² De-duplication refers to the process of locating and eliminating duplicate files.¹⁴³ It is

139. *In re Lorazepam & Clorazepate*, 300 F. Supp. 2d 43, 46 (D.D.C. 2004) (“the glory of electronic information is not merely that it saves space but that it permits the computer to search for words or ‘strings’ of text in seconds.”).

140. See Kenneth J. Withers, *Computer-Based Discovery in Federal Court Litigation*, 2000 FED. CT. L. REV. 2 (suggesting parties adopt collaborative strategies on search protocols); Robert D. Brownstone, *Collaborative Navigation of the Stormy e-Discovery Seas*, 10 RICH. J.L. & TECH. 53 (2004) (arguing that parties must agree to search terms and other selection criteria to narrow the scope to manageable data sets); *Treppel v. Biovail Corp.*, 233 F.R.D. 363 (S.D.N.Y. 2006) (showing defendant was justified in using keyword search terms to find responsive documents and should have proceeded unilaterally to use its proposed terms when the plaintiff would not agree); *Balboa Threadworks v. Stucky*, 2006 WL 763668 (D. Kan. Mar. 24, 2006) (ordering parties to meet and confer on the use of a search protocol, including key word searching).

141. “De-Duplication (‘De-Duping’) is the process of comparing electronic records based on their characteristics and removing or marking duplicate records within the data set. The definition of ‘duplicate records’ should be agreed upon, i.e., whether an exact copy from a different location (such as a different mailbox, server tapes, etc.) is considered to be a duplicate. De-duplication can be selective, depending on the agreed-upon criteria.” Sedona Conference Glossary, *supra* note 38.

142. At least two district courts appear to agree. *Medtronic Sofamor Danek, Inc. v. Michelson*, 229 F.R.D. 550, 552 (W.D. Tenn. May 13, 2003).

In the case of a large volume of data on multiple tapes like this case presents, the restored files from each tape must be compared to the restored files from every other tape and duplicate files eliminated. The restored files that are not duplicates must be converted to a common format so that a search program may seek information within them. The de-duplication and conversion are required so that large volumes of data in different formats may be searched in a reasonable time.

Id.; *In re CV Therapeutics, Inc. Sec. Litig.*, 2006 WL 2458720 (N.D. Cal. Aug. 22, 2006) (“Although Defendants complain that the resulting production and need for review of privileged matters is too burdensome, the permitted employment of de-duplication and search terms strikes a reasonable balance between Plaintiff’s needs and Defendants’ burden.”).

143. *L-3 Commc’ns Westwood Corp. v. Robicharux*, 2007 WL 756528, at 2 n.4 (Mar. 8, 2007) (giving a good example of the employment of hash analysis to locate and de-duplicate relevant files).

commonly utilized when reviewing computer files to remove exact duplicates from the review process.¹⁴⁴ Effective de-duplication of exact files is only possible by the use of hash. The process involves comparing the hash values of different ESI files and eliminating redundant files with identical hash values. This can drastically reduce the work of reviewers by sparing them from reading the same records over and over again.¹⁴⁵

Of course, as mentioned, a difference as small as a comma, or a new saved date, will render a completely different hash value, and so it is sometimes necessary to do “near de-duplication” analysis to reduce unproductive review.¹⁴⁶ Near de-duplication is the location and elimination of files wherein only segments of the file are identical to other files. It allows identifying similar but not identical files. This is still a developing area of technology, but several methods of near de-duplication have already been developed. For example, you can use a multiple hashing process to determine if the same content is contained in files with different fonts, or was saved in different file types, such as in Word, WordPerfect or Adobe PDF format. Hashing can also be used to determine when fields or segments within files are identical, even though the entire file might be quite different. It works by hashing only portions of a file. Thus, for instance, you can hash only the body of an e-mail to determine whether it is identical with another e-mail, even when the “reference” or the “to” and “from” fields are different.¹⁴⁷

Another variation of de-duplication is called “family hashing” where larger, logically related groups of files are hashed together.¹⁴⁸ Family hashing, also known as “family MD5 hash,” includes file metadata and both parent and attached files in the hash group. The hash of the entire group of related files is called the “family hash” value of the entire group. Thus, for instance, an e-mail and all of its associated attachments would be hashed together as one compound file. For two e-mails to be considered

144. Cohen & Lender, *supra* note 84.

145. *Id.*; Fordham, *supra* note 55, §§ 109-110. The chapter in Fordham’s book on Producing Electronically Stored Information, has a good description of MD5 hash and de-duplication, including valuable practical recommendations on: 1) the “granularity” of de-duplication with compound documents; and 2) maintaining a list of file hashes that includes the original file location path. As an example of the first “granularity” recommendation, an e-mail with attachments is considered a compound document. Fordham recommends that the e-mail itself be subject to one hash, and the attachments to additional separate hashes. Alternatively, they could all be hashed together to produce a hash value for the compound file.

146. *Id.* § 9.03[C].

147. *Id.*

148. *Id.* § 9.03[B].

identical, not only would the e-mail itself have to be identical, but also all of its attachments.¹⁴⁹

De-duplication is one of the best methods to save time and money in e-discovery projects. Hashing not only makes de-duplication and partial de-duplication possible, it makes it fast, cheap, and effective. Since most computers contain a high percentage of duplicate files, significant time savings can be realized in data harvesting, analysis, and production.¹⁵⁰ When considering millions of pages of data, this is not a luxury, it is a necessity.

The de-duplication of data restored from back-up tapes is even more important, especially where e-mails are concerned. On back-up tapes, the same saved ESI files can be included in every back-up: daily, weekly, monthly, and yearly. With e-mail, the redundancy is worsened because the same e-mail could have been sent multiple times to and from multiple people in the same organization. As a result, an individual can have hundreds of copies of the same e-mail.

This situation was discussed in an order resolving an e-discovery dispute in a class-action sexual harassment case.¹⁵¹ The plaintiff had complained about the reduction in the number of e-mails to be produced, from 17,375 to 8,660, because of the de-duplication performed by the defendant's e-discovery vendor, Kroll Ontrack. The Court upheld the reduction in production as valid and approved the de-duplication process with the following observation:

Kroll was also instructed to use the process of de-duplication, the process whereby documents which appear in a user's mailbox on multiple days are not counted as multiple hits. For example, if the same e-mail appeared in an inbox over a period of several months, only one copy of the document would be produced. After de-duplication, Kroll found 8,660 documents by searching for the 8 search terms, and by accounting for spam and family-cascading.¹⁵²

149. Fordham, *supra* note 55.

150. *Id.*; Dan Mares, *Using File Hashes to Reduce Forensic Analysis*, SC MAG. (Asia), May 1, 2002, at <http://scmagazine.com/asia/news/article/419780/> (last visited May 24, 2007).

151. *Wiginton v. CB Richard Ellis, Inc.*, 229 F.R.D. 568, 570-71, N.D. Ill. (2004).

152. *Id.*

Since today most employees routinely save e-mails and other electronic files, de-duplication, according to Kroll, can filter out as much as 60-80% of the files on backup tapes.¹⁵³

B. Known ESI Elimination

Hashing also allows for easy filtration of known files, such as operating system files and applications.¹⁵⁴ This is also sometimes referred to as a type of “data culling.”¹⁵⁵ Many if not most files on office computers are of this nature, and therefore have no possibility of containing relevant information.¹⁵⁶ Hashing allows an individual to easily filter out these standard files.¹⁵⁷ After filtering, an attorney will only have to review files for possible production that might contain relevant data, and will only have to look at them once, instead of multiple times.

VII. A MODEST PROPOSAL

Hashing is fast becoming a standard protocol in e-discovery.¹⁵⁸ A hash value is calculated for files at the time they are copied or “harvested” from a party’s computer system. The hash value is then attached to files as a field of a load file.¹⁵⁹ This system allows parties to track the hash values

153. Stuart Hanley, *E-Discovery “A to Z,”* Learning Lab; Presentation at West Legalworks/KrollOnTrack Continuing Legal Education Workshop (Oct. 2006). This estimate may be very conservative. *See, e.g., in re CV Therapeutics, Inc. Sec. Litig.*, 2006 WL 2458720, at *2 (N.D. Cal. Aug. 22, 2006) (showing where de-duplication of files restored from back-up tapes reduced the number of documents from 423,835 to 129,000, of which only 4,000 later proved to be responsive).

154. The National Software Reference Library (NSRL), operated by the National Institute of Science and Technology, maintains the hash values of all software commonly found on computers. In November 2006, the NSRL had hash values for 38,528,599 files. National Software Reference Library Web Site, available at <http://www.nsrl.nist.gov/> (last visited May 24, 2007). Tim Boland & Gary Fisher, Section of Hashing Algorithms, National Software Reference Library (June 30, 2000), at <http://www.nsrl.nist.gov/documents/hash-selection.pdf> (last visited May 24, 2007).

155. *See* EDRM Processing Stages, Data Culling, at http://www.edrm.net/wiki/index.php/Processing_-_Processing_Stages#Data_Culling (last visited May 24, 2007).

156. Fordham, *supra* note 55, at 112.

157. Hashing will also reveal if viruses or other malicious codes have altered the standard software files.

158. *See, e.g., ARKFELD, supra* note 88.

159. The same load file should also contain information as to the location on a party’s computer system from whence the file was harvested. In some instances, it may be necessary to prove who had what ESI on what computers. The load file could be consulted to trace the original location of any file, and other key chain of custody facts. *See, e.g., ARKFELD, supra* note 88, § 5.5(G).

of unique ESI files or groups of files, and thereby know whether they have been altered in any way. This procedure is becoming a de facto standard because it allows parties to verify the authenticity of any ESI at any time by simply running a hash calculation.¹⁶⁰

Since hash values are so important to ESI management, the hash characters unique to a computer file should be incorporated into a new naming convention for electronic records. This new naming convention should be used instead of Bates numbering in final ESI production and for use at depositions and trial.

First, the hash values should always be calculated and tied to each file at the time ESI is initially harvested from the party's computers.¹⁶¹ The next step in e-discovery is processing, which includes the weeding out of unresponsive files from the total universe of ESI collected. This filtration of ESI is accomplished by various methods, including de-duplication to eliminate matching hash values. Then the next step under the Electronic Discovery Reference Model standard, is to review these files, both by computer and manually.¹⁶² In the review process more files are eliminated as non-responsive or irrelevant.¹⁶³ Other documents will be screened at this review stage as privileged and logged.¹⁶⁴ The ESI left are those determined to fall within the scope of the party's initial production duties, or a later document request.¹⁶⁵ Then, when the ESI files are actually produced, the production should be accompanied by a hash value log that records all of the files produced and identifies them by hash value.

At this point in the process, it may be advantageous for the producing party to further name and identify the ESI files produced. Alternatively,

160. *Id.* This procedure was, for example, followed in *Williams II*. The hash values in the email and attachment load files could be used to match attachments to their transmitting e-mails and verify they had not been altered. For that reason the motion to compel a second production of the e-mails in native form was denied as unnecessary.

161. Collection is step four of the Electronic Discovery Reference Model Project. EDRM Project Web Site, *supra* note 137.

162. EDRM Project Web Site, *supra* note 137, step 6.

163. *Id.*; Sedona Conference Glossary, *supra* note 38 (defining "review" as follows: "Review: The culling process produces a dataset of potentially responsive documents which are then examined and evaluated for a final selection of relevant or responsive documents and assertion of privilege exception as appropriate.").

164. Sedona Conference Glossary, *supra* note 38 (defining "Privilege Data Set" as follows: "The universe of documents identified as responsive and/or relevant, but withheld from production on the grounds of attorney-client privilege or work product.").

165. This determination is part of step seven, "Analysis," in the Electronic Discovery Reference Model Project. EDRM Project Web Site, *supra* note 137.

166. "Production" is step eight in the Electronic Discovery Reference Model Project. *Id.*

the parties may wait to name particular ESI files until after they have been selected for actual use at depositions, hearings, or trial.

Regardless of whether the MD5 hash method is used with 32 digits, or the SHA-1 with 40 digits, both are too long for a practical naming convention. For instance, referring to a file or exhibit as Hash Number 5F0266C4C326B9A1EF9E39CB78C352DC does not work; it is too long and unwieldy to be practical. It takes too long to read such a number; the values are easily confused and misread, and it is impossible to memorize. For this reason, the author suggests the hash be truncated and only the first and last three places in the hash value be used, with a dot placed in between.¹⁶⁷

Thus, in the above hash, the characters to use would be 5F0.2DC. The full 32 characters of the hash fingerprint will still be preserved on the computer output file for reference purposes, but only the first and last 3 places, 6 out of 32 or 40, need be used in the name. In the rare event (little over one in a hundred) that the first and last three hash characters coincide between two ESI files, then the full hash values would be consulted.¹⁶⁸

167. A similar “truncation of hash” method is discussed in the patent case *Cable & Wireless Internet Services, Inc. v. Akamai Technologies, Inc.* where the court describes hashing and truncation as follows:

The patent describes several ways of determining the new name, including the use of a family of mathematical functions called message digest (“MD”) functions. One well known algorithm is MD5. The application of the MD5 function generates a very long unique identifier known as an MD5 hash value, or simply hash. Plaintiff suggests in its brief in support of the injunction that the number may be shortened or truncated for convenience by removing nearly half of the characters. Even as so reduced the number will remain “substantially unique” to the data so that if the data file is revised, application of the same MD5 function to the revised data will produce a different number. However, none of the asserted claims discusses truncating the hash.

Cable & Wireless Internet Serv., Inc. v. Akamai Techs., Inc., No. Civ.A.02-1 1430-RWZ, 2003 WL 1916691, at *1 (D. Mass. 2003). The case continues in this vein discussing an ingenious application of combined hashing to determine when web pages have been revised. The patent discussed in this case, and input I have received from patent lawyers, suggests that my model ESI hash naming proposal could be patented, but I prefer instead that it be freely disseminated as a noncommercial “open-source” idea. Indeed, I encourage the legal profession and e-discovery vendors to adopt and freely use this protocol or modify it to fit their purposes. My only claim here is to be the originator, and occasional mention thereof would be appreciated. As far as I know, I am the first to think of this truncated hash naming protocol, and know of no other similar proposals.

168. A match is only likely to occur approximately 1.4% of the time. In other words, on average, only one computer file in a hundred will have the same first and last three hash characters of a different file. This estimate is based on a study performed by Bill Speros, a computer expert

I further propose that the number sign (#)¹⁶⁹ affixed in front of the hash value abbreviation to make clear that a hash value follows. In every country of the world, except the United States, Great Britain, and other former British Commonwealth countries, the pound symbol (£) is known as the hash symbol.¹⁷⁰ So as a small step to positive globalization, I propose that American lawyers begin using the # to serve as a prefix for a computer hash number. Thus, the above would be written as #5F0.2DC.

It is important to have a # sign to demarcate where the hash characters begin, because the last prong of the proposed ESI naming convention is to put a more human moniker to the left, the way we currently do with automated Bates stamping. The name could be anything the litigant deems appropriate, as is the current custom. An individual could use letters, numbers, or both. As a default, but not necessarily a hard and fast rule, I propose that ESI be labeled in the left field with the name of the true author of the file. Thus, if an e-mail was written by Frank Jones, it would be labeled, "Frank Jones #A73.9B3." In cases where the author is unknown, the name of the custodian could be used.

Unlike a Bates stamp, a hash mark cannot be added to a native file directly because this would change the file and create an entirely new hash mark. However, you can modify the original name of the file to include the hash mark abbreviation because only the contents of a file are hashed, not the file name. Thus the original name of the computer file could be followed with the hash mark abbreviation; "Bill47.doc" would become "Bill47#A5D.7C1.doc." In addition, or alternatively, an individual could maintain the full hash value of a file by using a separate but linked load file. This obviates the need to revise the original name.

As yet another alternative, an individual could convert native files into TIFF, JPEG, or PDF files to facilitate further review, and eventual printing of certain records. After this conversion the full hash mark could be added to the face of each file by the conversion software, along with the author's

and attorney, who evaluated this aspect of the authors' proposal. He compared the MD5 hash codes of 460,477 files obtained from a typical manufacturing company's servers, plus several dozen of their networked and stand-alone PCs. Out of the 460,477 files checked, only 6,346 had the same first and last three characters. If the protocol was changed to the first and last four characters, there were only 24 matches. Bill Speros, *Private Correspondence*, E-mail from Bill Speros, an attorney in Cleveland, Ohio, with 19 years experience consulting in litigation technology and data management (Dec. 3, 2006) (on file with author).

169. See Wikipedia, *Number Sign*, http://en.wikipedia.org/wiki/Number_sign (defining the number sign and giving uses of it) (as of May 24, 2007 09:45 EST); Wikipedia, *Hash*, <http://en.wikipedia.org/wiki/Hash> (showing references to the word hash) (as of May 24, 2007 09:45 EST).

170. See *supra* note 169.

proposed shortened first and last three characters version, and, when desired, the name or number¹⁷¹ of the party's choosing. Then when the file is viewed on screen or in hard copy, the hash marks will appear also.

This conversion and hash marking would necessarily be done prior to the time of production of ESI, much like the way Bates stamping is currently done. In any event the hash marking should be completed before the ESI files are presented¹⁷² by use at deposition, hearing, or trial.

Regardless of the timing, procedure, or specific methods of deployment, the key to the successful implementation of a hash naming protocol is good communication with opposing counsel, and, if possible, agreements on the procedures to be followed.¹⁷³ Ideally, such procedures would be discussed in the initial "meet and greet" session of counsel, as part of the discussion and agreement on the form of production.¹⁷⁴ Counsel should also exchange hash lists and naming protocols before depositions or hearings, or at least, when the depositions or hearings begin. This would allow all parties to follow along and verify that the submitted ESI was not altered. Objections may be made based upon an instant hash of a computer file that shows it has been altered.

When the files have not been submitted to opposing counsel before the deposition or hearing, the hashing will have to be made on the fly during the proceeding. Thus, when an electronic file is shown to a witness on screen, it may become routine for opposing counsel, or the witness herself, to make a quick hash check of that file before the witness accepts it as authentic. If not, objections and *voir dire* on the issue may be appropriate. Of course, this would require all counsel to have computers with them, hash software, and the submission of files via CD or direct connection. Although to the author's knowledge this has never yet occurred, it is likely to become commonplace in five to ten years.

At trial, full advance disclosure of all exhibits to be used is generally required, along with pre-marking of exhibits. The further identification of trial exhibits should include the hash characters and the naming protocol recommended here. Since trial exhibits are pre-marked with full disclosure, the pressure of instant hashing of ESI shown to witnesses at

171. The original sequential Bates number may yet survive for a few years as lawyers and vendors continue to use them in this modified manner.

172. Presentation is the final ninth step in the Electronic Discovery Reference Model Project. EDRM Project Web Site, *supra* note 137.

173. See, e.g., *supra* notes 89 & 140.

174. FED. R. CIV. P. 16(b), 26(f); see also *supra* note 89; *Hopson v. Mayor of Baltimore*, 232 F.R.D. 228, 245 (D. Md. 2006).

trial should be lessened. ESI will have all been pre-authenticated, or at least subject to review and objections, or motions in limine.

In countless courtrooms today, a mantra something like this is heard often: "I am handing the witness a document pre-marked as 'Trial Exhibit 75' and Bates stamped as 'Dr. Smith 0573.'" In the future, the author expects something like this will be heard instead: "I am putting on screen for the witness to view an ESI file pre-marked as 'Trial Exhibit 75' and hash marked as 'Dr. Smith Hash 4F7.C3B (Dr. Smith#4F7.C3B).'" The ESI file may still sometimes be converted to paper, in which case it could be handed to a witness, instead of put on a screen, but the same naming protocol would apply and it would bear a "hash mark" somewhere on the bottom: "Dr. Smith#4F7.C3B."

Sorry, Mr. Bates, your one hundred-year-plus reign is over.